

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
Теплоенергетичний факультет
Кафедра автоматизації проектування енергетичних процесів і систем

До захисту допущено:
Завідувач кафедри
_____ Олександр КОВАЛЬ
«__» _____ 2020 р.

Дипломна робота
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Інформаційні технології
моніторингу довкілля»
спеціальності 122 «Комп'ютерні науки та інформаційні технології»
на тему: «Програмна платформа для організації міжнародних
конференцій»

Виконав:
студент IV курсу, групи ТМ-62
Гладкий Олег Леонідович

Керівник:
Доцент, кандидат технічних наук
Кублій Лариса Іванівна

Рецензент:
Доцент, кандидат технічних наук
Самарай Валерій Петрович

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.
Студент _____

Київ – 2020 року

Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти перший (бакалаврський)

Напрямок підготовки 122 Комп'ютерні науки та інформаційні технології

Спеціалізація Інформаційні технології моніторингу довкілля

ЗАТВЕРДЖУЮ

Завідувач кафедри

Олександр КОВАЛЬ
(підпис)

” ____ ” _____ 2020р.

ЗАВДАННЯ

на дипломну роботу студенту

Гладкому Олегу Леонідовичу _____

1. Тема роботи “Програмна платформа для організації міжнародних конференцій”

керівник роботи Кублій Лариса Іванівна, к.т.н., доцент

(прізвище, ім'я, по батькові науковий ступінь, вчене звання)

затверджена наказом вищого навчального закладу від “25” травня 2020 р.
№ **1168-с**

2. Строк подання студентом роботи 14 червня 2020 р.

3. Вихідні дані до роботи персональний комп'ютер під керуванням операційної системи Microsoft Windows, мова програмування TypeScript, мова програмування NodeJs.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) проаналізувати існуючі програмні рішення та можливі засоби реалізації взаємодії, обґрунтувати вибір програмних застосунків і шляхи розробки програмних додатків, розробити програмне забезпечення, розробити користувацький інтерфейс

5. Перелік ілюстраційного матеріалу (з точним зазначенням обов'язкових креслень) 1. Завдання розробки програмної платформи для організації міжнародних конференцій. 2. Існуючі веб-платформи для організації міжнародних конференцій. 3. Засоби розробки програмної платформи. 4. Архітектура веб-платформи. 5. Сторінка перегляду конференції користувачем. 6. Сторінка для оновлення інформації. 7. Висновки

6. Публікації: Матеріали XVIII Міжнародної науково-практичної конференції молодих вчених і студентів 2020 року “Сучасні проблеми наукового забезпечення енергетики”, м. Київ, КПІ ім. Ігоря Сікорського.

7. Дата видачі завдання ”15” __жовтня__ 2019_ р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів виконання дипломної роботи | Термін виконання етапів роботи | Примітки |
|-------|---|--------------------------------|----------|
| 1. | Затвердження теми роботи | 03.12.19 | |
| 2. | Вивчення та аналіз задачі | 13-19.04.20 | |
| 3. | Розробка архітектури та загальної структури системи | 20-26.04.20 | |
| 4. | Розробка структур окремих підсистем | 27.04-03.05.20 | |
| 5. | Програмна реалізація системи | 04-13.05.20 | |
| 6. | Оформлення пояснювальної записки | 14-16.05.20 | |
| 7. | Захист програмного продукту | 17.05.20 | |
| 8. | Передзахист | 09.06.20 | |
| 9. | Захист | 16.06.20 | |

Студент

(підпис)

Гладкий О. Л.

(прізвище та ініціали)

Керівник роботи

(підпис)

Кублій Л. І.

(прізвище та ініціали)

АНОТАЦІЯ

Мета роботи — розробка програмної платформи для організації міжнародних конференцій. У процесі виконання роботи досліджено принципи веб-програмування та можливості їхньої реалізації з використанням технологій HTML, CSS, TypeScript, React, Node.js, MongoDB.

Результатом роботи є веб-платформа, написана за допомогою середовища розробки Microsoft Visual Studio Code. Записка містить 85 сторінок, 27 рисунків, 31 посилання і 4 додатки.

Ключові слова: конференція, реагувати, подія, google, mongodb.

ABSTRACT

The purpose of the work is to develop a software platform for the organization of international conferences. In the process of work the principles of web programming and possibilities of their realization with use of technologies HTML, CSS, TypeScript, React, Node.js, MongoDB are investigated.

The result is a web platform written using the Microsoft Visual Studio Code development environment. The note contains 85 pages, 27 figures, 31 links and 4 appendices.

Keywords: conference, react, event, google, mongodb.

ЗМІСТ

| | |
|---|----|
| Перелік умовних позначень, скорочень і термінів | 7 |
| ВСТУП | 8 |
| 1. ЗАВДАННЯ РОЗРОБКИ ПРОГРАМНОЇ ПЛАТФОРМИ ДЛЯ ОРГАНІЗАЦІЇ МІЖНАРОДНИХ КОНФЕРЕНЦІЙ | 10 |
| 1.1 Задача планування та організація заходу за допомогою веб-платформи | 10 |
| 1.2 Задача вибору програмного забезпечення для організації заходів | 11 |
| 1.3 Аналіз наявного програмного забезпечення для організації заходів | 12 |
| 1.3.1 Сегментація | 14 |
| 1.3.2 Регіональний аналіз | 15 |
| 1.4 Визначення переваг та необхідності програмного забезпечення для організації конференцій | 15 |
| 2. ІСНУЮЧІ ВЕБ-ПЛАТФОРМИ ДЛЯ ОРГАНІЗАЦІЇ МІЖНАРОДНИХ КОНФЕРЕНЦІЙ | 18 |
| 2.1 Існуючі платформи для організації заходів | 18 |
| 3. ЗАСОБИ РОЗРОБКИ ПРОГРАМНОЇ ПЛАТФОРМИ | 26 |
| 3.1 Мова програмування TypeScript | 26 |
| 3.2 Бібліотека для створення інтерфейсу React | 28 |
| 3.3 Мова запитів GraphQL | 29 |
| 3.4 База даних MongoDB | 29 |
| 3.5 Сервіс Cloudinary | 32 |
| 3.6 Редактор коду Microsoft Visual Code | 33 |
| 4. РОЗРОБКА ПЛАТФОРМИ | 34 |
| 4.1 Архітектура веб-платформи | 34 |
| 4.2 Створення сторінки перегляду конференцій користувачем | 36 |
| 4.3 Розробка сторінки для створення нової конференції | 37 |
| 4.4 Розробка сторінки для оновлення інформації | 38 |

| | |
|--|----|
| | 6 |
| 4.5 Розробка сторінки квитків на конференцію | 40 |
| 5. ОПИС РОБОТИ КОРИСТУВАЧА З ПЛАТФОРМОЮ | 42 |
| 5.1 Системні вимоги | 42 |
| 5.2 Використання платформи | 42 |
| ВИСНОВКИ | 47 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ | 48 |
| Додаток А | 51 |
| Додаток Б | 53 |
| Додаток В | 74 |
| Додаток Г | 82 |

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

JavaScript — браузерна мова програмування

ReactJS — фреймворк мови JavaScript для створення користувацького інтерфейсу

Node.js — платформа для створення серверної частини

GraphQL — мова для створення та спрощення запитів

MongoDB — сервер бази даних

EMS — сервіс доставки електронної пошти

CDN — сервіс доставки вмісту

IDE — інтегроване середовище розробки

ВСТУП

Кількість користувачів браузерних веб-додатків за останні роки значно зросла. Організатори різноманітних заходів усвідомили, що веб-додатки неминуче піднімають проведення заходів на новий рівень. Для відвідувачів вони зручні й привабливі, організаторам дають можливість легко збирати дані і створювати звіти. Друковані версії рекламних матеріалів і розкладів залишаються в минулому.

Програмне забезпечення для організації конференцій або інших заходів — це засіб, який допомагає організаторам досягнути успіху в усіх аспектах — від реєстрації і планування заходів до створення звітів. Оскільки програмне забезпечення для керування заходами об'єднує всі інструменти в одну платформу, то це робить програму легкою для розуміння і надає організаторові велику кількість готових ідей.

За останнє десятиліття додатки для проведення заходів перетворилися в універсальний інструмент для планувальників і відвідувачів. Вони роблять відвідування заходів більш продуктивними.

Інноваційні технології відкривають нові можливості для організаторів заходів. Поява нових програмних платформи дала можливість обміняти багатосторінкові друковані плани заходу на програмне забезпечення на комп'ютерах. Основні функції додатків змінилися, пропонуючи відвідувачам нові способи взаємодії зі змістом конференції і можливість спілкуватися зі своїми колегами. Такі переваги платформи допоможуть залучити відвідувачів до заходу. Деякі з переваг використання програми як програмного продукту подій зрозумілі й необхідні. Якщо відбуваються зміни в останню хвилину, інформація про захід може бути легко оновлена і миттєво поширена серед усіх учасників. Після того, як весь процес реєстрації подій перемістився в онлайн, в організаторів є детальна база даних учасників події. Це виключає ймовірність помилок або розбіжностей під час реєстрації.

Автоматизація процесу організації та проведення міжнародних наукових конференцій полягає в створенні програмної платформи для організації міжнародних конференцій. У програмній платформі представлена можливість створення, редагування і видалення інформації про конференції, запрошення на конференцію шляхом відправлення відповідної інформації на пошту, завантаження файлів будь-якого типу та розміщення їх на головній сторінці конференції. Наявність такої платформи має спростити процес для організаторів конференції.

Для розробки програмної платформи обраний Visual Studio Code — редактор коду, мову програмування — TypeScript. Для зберігання даних використано MongoDB — базу даних і сервіс зберігання завантажених файлів Cloudinary. Програма може бути запущена на будь якій операційній системі завдяки сервісу для контейнеризації Docker.

Розроблена програмна платформа призначена в першу чергу для організаторів наукової міжнародної конференції та її учасників.

У першому розділі дипломної записки зроблено постановку задачі. У другому розділі проаналізовано існуючі програмні рішення. У третьому розділі обґрунтовано використання засобів розробки. Темою четвертого розділу є програмна реалізація, де детально подано логічну структуру веб-порталу. У п'ятому розділі розглянуто можливості роботи користувача з програмною платформою.

1. ЗАВДАННЯ РОЗРОБКИ ПРОГРАМНОЇ ПЛАТФОРМИ ДЛЯ ОРГАНІЗАЦІЇ МІЖНАРОДНИХ КОНФЕРЕНЦІЙ

Організація будь-якої конференції — досить складне завдання. Проте за допомогою ретельно розробленого плану і правильно обраних інструментів конференцію можна організувати і провести без особливих складнощів.

1.1 Задача планування та організація конференцій за допомогою веб-платформи

Планування конференції є складним завданням. Потрібно визначити дату конференції, скласти інформаційний лист, визначити і оформити місце проведення конференції, зібрати заявки потенційних учасників, тексти їхніх доповідей, презентації, інший матеріал, інформацію про оплату підготовки й видання матеріалів, спілкуватися з учасниками, розмістити різноманітні оголошення, сформулювати за інформацією з заявок програму, розмістити її на сайті і розіслати учасникам, розмістити на сайті звіт конференції, включаючи фотографії й відеоматеріали, збірник матеріалів конференції.

Наявність при цьому програмного забезпечення для організації конференцій робить процес швидким і простим для організаторів, а також доступним для учасників конференції і всіх зацікавлених осіб. Організатор конференції позначає захід у календарі з урахуванням точного часу. Він також повинен мати розширену систему повідомлень для забезпечення того, що користувач ніколи не пропустить захід, який він хоче відвідати. Тому такий додаток повинен мати вбудований календар з налаштованими повідомленнями. Деякі додатки для event-менеджера можуть інтегруватися з Google Календарем і дати можливість користувачеві планувати свій графік, не закриваючи браузер [1]. Додаток має надавати користувачам тільки актуальну інформацію. Розклад заходів повинен регулярно

оновлюватися. Додаток для заходів повинен містити ще одну важливу “соціальну” функцію. Багато заходів проводяться на регулярній основі — щорічно або навіть щомісяця. Ті, хто збирається відвідати конкретний захід вперше, хочуть знати, що інші люди, які вже були присутні на ньому, думають про цю конференцію або семінар [2].

1.2 Задача вибору програмного забезпечення для організації заходів

Якщо для підготовки і організації конференцій не використовувати спеціальне програмне забезпечення, то виникає досить складна ситуація: багато хто використовує велику кількість документів Word чи можливо Excel для управління інформацією, яка дублюється між ними. Коли учасники змінюють свої плани, змінюються імена або відбуваються скасування заходів, виникає плутанина при оновленні кількох документів [3]. Таким чином, організація заходів без спеціального програмного забезпечення може створити багато додаткової роботи.

Використання програмного забезпечення може допомогти організатору максимально збільшити кількість реєстрацій, одночасно перенести їх до мінімуму ресурсів. Хороші рішення допоможуть автоматизувати більшість процесів, пов'язаних з організацією і проведенням конференцій. Вони пропонують такі функції, як відстеження реєстрації, складання програми конференції, збір заявок на використання технічних засобів для доповідей, формування звітів конференції тощо. Можливо швидко відстежити інформацію про те, що відбувається, хто зареєструвався, хто робить доповідь. Всього за кілька натискань миші організатор може згенерувати звіт про проведений захід. Програмне забезпечення для управління заходами розробляється і створюється, щоб зробити планування швидшим і ефективнішим. Користувальницький інтерфейс цього програмного забезпечення має бути досить зрозумілим і простим для освоєння.

Організатори та відвідувачі заходів мають набір інструментальних панелей з великою кількістю функцій, що відносяться до різних аспектів планування з точки

зору цілої веб-платформи. Потрібно заповнити кілька форм, налаштувати дату і час проведення — і все готово. Ще однією з перевагою програмного забезпечення для проведення заходів є те, що організаційні робочі процеси доступні в системі і можуть бути адаптовані до конкретних потреб. Деякі процеси можуть бути навіть автоматизовані. Добре організований призначений для користувача інтерфейс допомагає відслідковувати найбільш актуальні деталі.

Організаторам заходів також складно відстежувати і координувати свої реєстрації та створювати належну базу даних учасників. Також можна відстежити покрокове оформлення заявки на сайті заходу.

Найбільшою перевагою використання платформи для організації заходу є те, що інформація може бути миттєво розповсюджена між учасниками і може оновлюватися в режимі реального часу. Всі інструменти для організації заходу обмінюються інформацією через єдину базу даних, що забезпечує актуальність інформації і даних незалежно від того, де вони знаходяться в життєвому циклі подій. Програмні рішення для планування заходів дозволяють легко планувати, організовувати і створювати цілісні заходи, які приносять результати. Організація заходів — це набагато більше, ніж просто планування заходів та розсилка запрошень [4]. Правильні інструменти можуть призвести до більшого успіху і кращих відносин.

1.3 Аналіз наявного програмного забезпечення для організації заходів

Університети регулярно проводять наукові конференції, тому для успіху цих заходів потрібно добре спланована реєстрація заходу, вибір місця проведення, керування учасниками, маркетинг подій і правильний вибір технологій [5]. Таке планування заходів вимагає багато часу підготовки, попереднього бронювання місця та ведення обліку у вигляді документів. Тому, складно управляти всіма цими завданнями незалежно.

Програмне забезпечення для управління подіями надає всі рішення, пов'язані з плануванням, з єдиної платформи. Планувальник подій може реєструвати події, може відправляти на електронну пошту запрошення та детальну інформацію про подію, може проводити опитування, щоб зрозуміти відгуки, і може створювати налаштоване середовище відвідувачів для взаємодії з учасниками, а також з іншими аудиторіями.

Це допомагає професіоналам при проведенні багатьох успішних заходів. Найпопулярнішим додатком для управління подіями є створення спеціалізованих веб-сайтів про події, онлайн-реєстрація, онлайн-платежі, вибір місця проведення, управління контентом, планування, створення мереж, а також створення онлайн-опитувань для отримання відгуків аудиторії [6].

Додатки Cvent, RegOnline, etouches, Bizzabo, ThnderTix, Eventmobi і EventsAir, є одними з популярних програм управління подіями, доступних на ринку клієнтів. Cvent, яка недавно об'єдналася зі своїми дочірніми рішеннями Lanyon, є провідним гравцем на ринку індустрії програмного забезпечення для управління подіями, за яким слідують etouches, EventsAir, Bizzabo і Ungerboeck International. Всі компанії пропонують програмне забезпечення для управління подіями, пов'язане з одним завданням, тоді як Ungerboeck international і Cvent пропонують менш 10%, які можуть надати всі функції управління подіями на одній програмній платформі [7].

Очікується, що все більша кількість додатків для управління подіями у браузері буде стимулювати ринок EMS, оскільки воно пропонує прості в управлінні функції і інструмент аналізу для оцінки рентабельності інвестицій, що створюються в галузі. Зростаючі інтернет-послуги на браузерній платформі сприяють зростанню кількості користувачів веб-додатків по всьому світу. Згідно з дослідженням MRFR більше 2 мільярдів чоловік користуються веб-додатками з особистою і професійною метою [8]. Така активна участь комп'ютерних пристроїв привертає все більше розробок в індустрії веб-додатків, оскільки це дуже ефективний метод для організації, що дозволяє охопити максимальну кількість людей одночасно.

При розробці програмних засобів організації конференцій треба також зважати на зростаючу кількість кібератак, що загрожують конфіденційності та

безпеці даних. Це є однією з проблем для користувачів, що пропонують програмні рішення для управління подіями, які можуть обмежувати зростання ринку в найближчі роки [9].

1.3.1 Сегментація

За типом програмного забезпечення ринок був розділений на аналітичне програмне забезпечення, заходи, реєстрацію, маркетинг подій, локальні технології, пошук об'єктів та інші.

За розгортанням ринок був розділений на локальні і хмарні.

За заявками, ринок був розділений на корпоративні, освітні, асоціативні, державні та інші.

Ринок глобального програмного забезпечення для управління подіями, за оцінками, буде зростати значними темпами протягом прогнозованого періоду з 2019 по 2025 рік (рисунок 1.1).

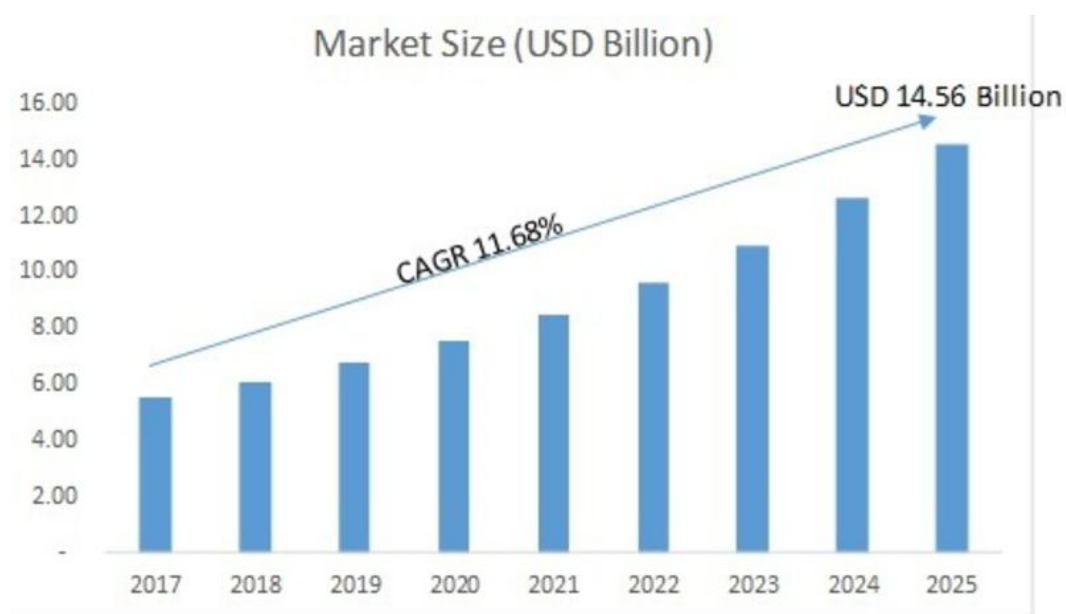


Рисунок 1.1 — Глобальний ринок програмного забезпечення для організації подій [10]

Географічний аналіз ринку програмного забезпечення для керування подіями вивчався для Північної Америки (США, Канада і Мексика), Європи (Німеччина,

Великобританія, Франція, Іспанія та інші країни Європи), Азіатсько-Тихоокеанський регіон (Австралія, Сінгапур, Нова Зеландія, Індія і інші країни Азіатсько-Тихоокеанського регіону), а також Близький Схід і Африка [11].

1.3.2 Регіональний аналіз

Високі темпи зростання ринку програмного забезпечення для управління подіями спостерігаються в Північній Америці. Присутність великих компаній з управління подіями, таких як Cvent, etouches, Ungerboeck, Bizzabo і інших, зростаюча потреба в технічно досконалому рішенні для обслуговування швидкозростаючої галузі управління подіями і успішна реалізація різних типів програмного забезпечення для управління подіями з використанням хмарного розгортання є основними факторами, що сприяють зростанню цього ринку в регіоні [12].

Індустрія управління подіями в Азіатсько-Тихоокеанському регіоні демонструє повільні темпи зростання протягом прогнозованого періоду. Нестача знань про технічні можливості й переваги програмного забезпечення для управління подіями, а також дотримання традиційних методів планування подій — головна причина такого повільного зростання в регіоні.

Останнім часом відбулися злиття і поглинання серед ключових учасників — стратегія, яку бізнес-суб'єкти використовують, щоб розширити охоплення своїх клієнтів.

1.4 Визначення переваг і необхідності програмного забезпечення для організації конференцій

Основною причиною будь-яких інновацій завжди були потреби і проблеми, які виникали. Використовуючи певні технології, вчені й спеціалісти кожної сфери знаходять способи зробити речі простіше і швидше. Галузь організації конференцій теж зіштовхнулася з цим. Виникало багато проблем планування заходів, які вимагали швидкого і простого рішення. Перш за все, виконання безлічі завдань

вручну займало багато часу і потребувало багато коштів. Крім того, зі збільшенням частоти проведення конференцій періоди планування скорочувалися. Відповідно, більш короткий час виконання замовлення, що збігається з розвитком технологій, призвели до розробки програмного забезпечення для організації та управління конференціями.

Функціями програмного забезпечення для управління подіями є:

- реєстрація події, детальний опис;
- оформлення основної сторінки конференції;
- організація заходу;
- управління відвідувачами;
- зворотній зв'язок;
- аналітика подій, щоб відстежувати все це.

З усіма цими функціями, можна визначити, що програмне забезпечення для управління конференції — це необхідність. Коли справа доходить до використання програмного забезпечення для управління подіями для створення заходу, кожне рішення знаходиться в руках планувальника подій. Користувач сам вирішує, яку інформацію слід відображати на головній сторінці створеного заходу.

Замість координації дій між кількома людьми в ієрархії, при використанні веб-сайту заходи, стратегії просування, зворотній зв'язок тощо — кожна частина процесу знаходиться під контролем організатора. Кожен організатор заходу хоче, щоб його конференція виділялася, справляла враження. Таким чином, одне з питань, що виникають у свідомості планувальника подій при виборі такого програмного забезпечення, полягає в тому, як вирізнити подію серед інших, створених з використанням того ж програмного забезпечення. Проста відповідь: налаштування та ручне створення інтерфейсу.

Програмне забезпечення для управління подіями залишає багато місця для налаштування: від розміщення логотипу заходу або використання колірної схеми кожен елемент заходу налаштовується. Організатор може оформити свою головну сторінку конференції і адаптувати її відповідно до вимог.

Використовувана колірна схема, розміщення логотипу, головна сторінка створеного заходу, кожен цифровий аспект заходу відображає почерк організатора. Замість того, щоб вибирати окремі платформи для кількох завдань, таких як реєстрація, створення веб-сайту і додатки для проведення заходів, розсилки електронною поштою, організатор може управляти всім цим з одного джерела.

Як уже вказувалося, програмне забезпечення для керування подіями було створено, щоб зробити планування швидшим і ефективнішим. Саме тому загальний користувацький інтерфейс цього програмного забезпечення досить легко освоїти. Часто планувальнику подій просто досить одного ознайомлювального виклику, щоб освоїти роботу з програмним засобом.

Планувальники подій мають набір інструментальних панелей з великою кількістю функцій, що відносяться до різних аспектів планування з точки зору веб-сайту подій, мережевої платформи та аналітики подій. Потрібно просто заповнити деяку попередню інформацію, налаштувати її так, як потрібно. Організатор може легко створити візуально привабливу головну сторінку конференції. Все це робиться всього за кілька натискань кнопок.

Складність реєстрації може бути додатково зменшена за допомогою QR-коду. Спосіб шифрування QR-код — це в основному двовимірний штрих-код, який можна сканувати. Кожному учаснику може бути присвоєно QR-код, який можна відсканувати під час реєстрації. Організаторам надається додаток організатора подій, щоб зробити цифрову реєстрацію ще більш зручною. Це прискорить реєстрацію і спростить управління учасниками. Ще однією перевагою цифрової реєстрації є те, що вона допомагає в забезпеченні безпеки подій.

2. ІСНУЮЧІ ВЕБ-ПЛАТФОРМИ ДЛЯ ОРГАНІЗАЦІЇ МІЖНАРОДНИХ КОНФЕРЕНЦІЙ

Веб-платформа для організації міжнародних конференцій повинна мати можливість створення, редагування, публікації конференцій. Кожна конференція як сутність характеризується наступними параметрами: унікальний ідентифікатор; назва; організатор; статус; тип проведення конференції; місце проведення; дата початку конференції; дата закінчення конференції; головне зображення конференції. Конференції можуть мати однакові назви, але вони розрізняються по своєму унікальному ідентифікатору (MongoID).

Статус конференції може бути: draft (незареєстровані користувачі не мають можливості побачити конференцію на головній сторінці); published (будь-який користувач може отримати доступ до сторінки конференції з головної сторінки).

Передбачаються певні обмеження на інформацію в системі. Наприклад, конференція може не мати жодного учасника; опис може бути не більше 600 символів; можливо завантажити тільки одне тематичне зображення.

При роботі з платформою користувач повинен мати можливість розв'язувати такі завдання: створювати, редагувати і публікувати конференції, прикріплювати додаткові матеріали, встановлювати тематичне зображення конференції, додавати та редагувати матеріали конференції.

2.1 Існуючі платформи для організації заходів

Для організації конференцій на сьогоднішній день розроблено досить багато програмних систем. Серед них можна виділити такі найпопулярніші, як веб-додатки Monday, EventBookings, Cvent, Eventzilla та хмарну платформу Gather.

Веб-додаток Monday [13] — відомий інструмент управління заходами, який дає можливість організовувати, керувати та стежити за своїми подіями та іншими пов’язаними завданнями.

У системі досить легко додати інформацію про бронювання, бюджет, відвідуваність та навіть квитки до кожного проекту події (рисунок 2.1). Якщо потрібно побачити “велику картину” або збільшити масштаб для детального перегляду, це можна легко зробити за допомогою даної системи. Ефективніше відстежувати терміни з наміченими датами та основними віхами.

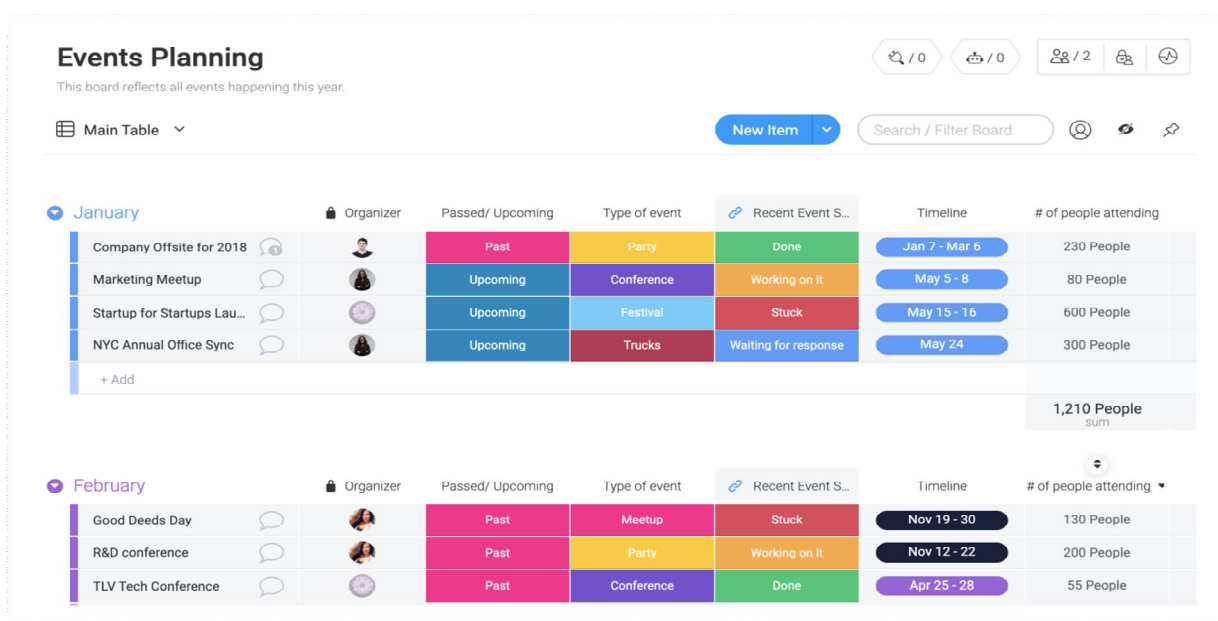


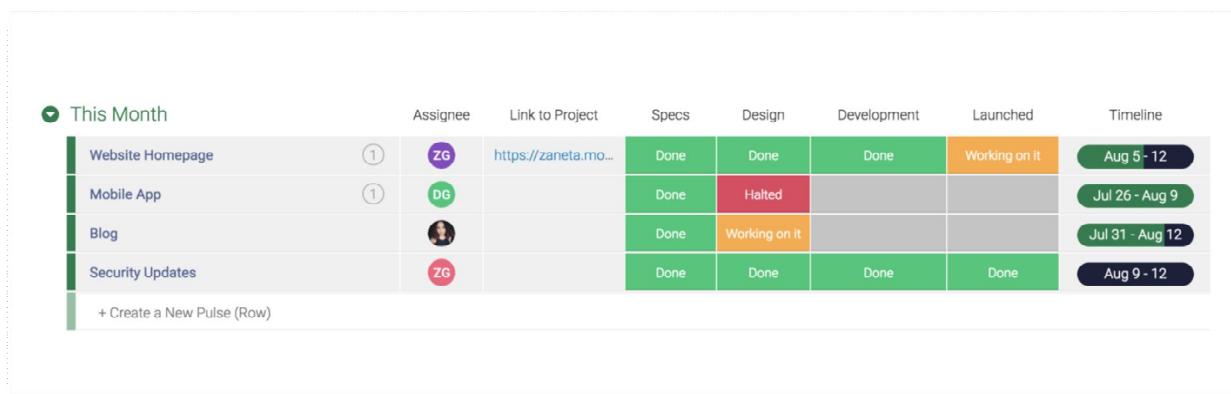
Рисунок 2.1 — Планування заходу в додатку Monday [13]

Потужна візуалізація платформи дає можливість отримати інформацію з першого погляду.

Завдання відстеження різних подій можуть бути складними та заплутаними. Додаток Monday може допомогти створити структуру та організувати ці завдання відповідно до пріоритетів, важливості і проектів.

Користувачі можуть спілкуватися в рамках заходу, тому легко вести дискусії для швидкого прийняття рішень. Додаток Monday надає командам можливість створювати власні робочі програми для будь-якого робочого процесу. Він може

використовуватися для чого завгодно — для команд, відділів, керівників та організацій та для всіх типів проектів і процесів (рисуюнок 2.2).



The screenshot displays the Monday.com interface for a project titled 'This Month'. It features a table with columns for Assignee, Link to Project, Specs, Design, Development, Launched, and Timeline. The table lists four items: Website Homepage, Mobile App, Blog, and Security Updates. Each item has a status indicator (e.g., 'Done', 'Working on it', 'Halted') and a timeline bar showing dates.

| | Assignee | Link to Project | Specs | Design | Development | Launched | Timeline |
|------------------|----------|---|-------|---------------|-------------|---------------|-----------------|
| Website Homepage | ZG | https://zaneta.mo... | Done | Done | Done | Working on it | Aug 5 - 12 |
| Mobile App | DG | | Done | Halted | | | Jul 26 - Aug 9 |
| Blog | | | Done | Working on it | | | Jul 31 - Aug 12 |
| Security Updates | ZG | | Done | Done | Done | Done | Aug 9 - 12 |

+ Create a New Pulse (Row)

Рисуюнок 2.2 — Веб-додаток Monday [13]

Для кожної таблиці можна додавати колонки відповідно до робочого процесу. Веб-додаток Monday пропонує кілька типів колонок, щоб відповідати потребам користувачів, від колонки прізвищ учасників до колонки статусу.

Веб-додаток EventBookings [15] робить планування подій та оформлення квитків максимально швидкими й надійними. Після реєстрації можливо відразу перейти до створення сторінки події, де можна придбати квитки та збирати дані про гостей. Через платформу можна приймати платежі за допомогою кредитних і дебетових карток та через Stripe.

Платформа дає можливість розміщувати власний брендинг на сторінках подій, включаючи власні кольори та логотипи. Крім того, можливо запропонувати кілька рівнів ціноутворення та легко коригувати ціни за потребою. Система також дає можливість встановити знижки на квитки, щоб заохотити відвідування події за допомогою вбудованих можливостей для раннього ціноутворення та код купонів.

Після того як користувач поділився посиланням на свою сторінку подій, на інформаційній панелі EventBookings відображаються такі показники, як продані квитки, унікальні відвідувачі.

Платформа також пропонує програму для сканування, яку можна використовувати в день своєї події для реєстрації гостей. Додаток надає можливість друкувати списки гостей, а не використовувати програму сканера.

На платформі можливо створити кілька рівнів ціноутворення та регулювання ціни на квитки відповідно до попиту на подію (рисунок 2.3). Існує можливість створення квитка на основі аудиторії, яка приходить на створену подію.

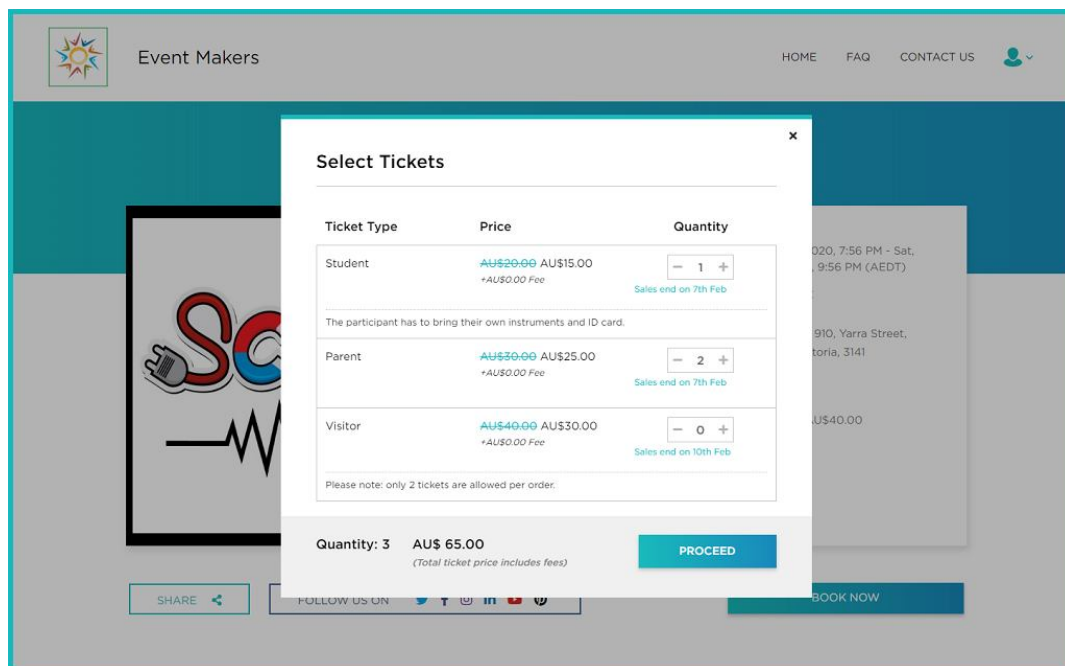


Рисунок 2.3 — Можливість створення квитків [15]

Це особливо добре спрацює, коли користувач хоче показати дешеві квитки як велику пропозицію та надавати більш дорогі квитки. Найкращі місця в приміщенні або спеціальний доступ можна отримати тим, хто має конкретну категорію квитка.

Хмарна платформа Gather [16] — це платформа керування подіями, що добре працює для підприємств усіх рівнів, які хочуть організувати будь-який захід. Від планування до етапу запуску ця платформа може допомогти організаторам забезпечити безперебійність всього процесу для всіх (рисунок 2.4). Незалежно від того, яким з яким пристроєм працює користувач, в системі можливо синхронізувати свої дані та оновлення календаря з іншою частиною команди через хмару, щоб усі могли залишатися на одній сторінці.

Хмарний веб-додаток Gather дає можливість користувачам надсилати запити, бронювати події в будь-який час через віджет бронювання або планувати події.

Після проведення заходу, кілька натискань на пристрої дають можливість надіслати рахунок-фактуру та звіт. Через Gather користувачі зможуть без жодних труднощів надсилати свої платежі, оскільки додаток забезпечує безпеку та безпеку фінансових реквізитів, а також відповідає сумісності PCI DSS.

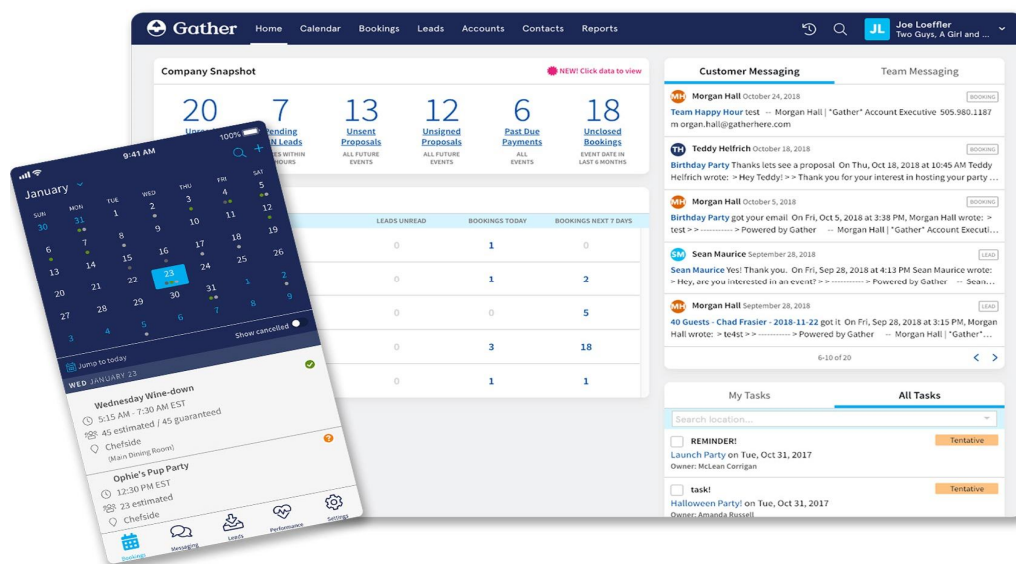


Рисунок 2.4 — Головна сторінка платформи Gather [16]

Ця платформа пропонує такі функції, як автоматичні мобільні сповіщення та оновлення, які дають можливість власникам відстежувати графіки роботи і заходи подій. Спільний календар і функціональні можливості обміну повідомленнями команди надають можливість командам координувати дії і співпрацювати з будь-якого місця.

Автоматичне зберігання допомагає збільшити кількість потенційних клієнтів за допомогою вбудованого віджета запиту бронювання, щоб автоматично зберігати інформацію про потенційних клієнтів у будь-який час дня і ночі. Час відповідей скорочується за допомогою таких можливостей, як цифрові пропозиції та електронний підпис.

Веб-додаток Cvent [17] — це комплексне рішення, стійке до потреб організаторів за галузями, роллю і навіть для сторонніх планувальників. Він має унікальний інструмент для пошуку найбільш рентабельного місця для

організованого заходу. Плани масштабуються до малих і великих подій та охоплюють весь життєвий цикл подій, починаючи від планування і закінчуючи звітуванням.

Додаток Cvent має гнучкі засоби комунікації, які дають можливість користувачеві об'єднувати учасників у групи і робити спілкування електронною поштою більш цільовим і ефективним. Також присутнє налаштування інтерфейсу для проектування заходу. Для планувальників-фрілансерів ця здатність надає їм професійного вигляду, що виправдовує більш високу плату за консультації.

Так само автоматизація допомагає команді зосередитись на маркетинговому і творчому процесах, а не перенавантажуватися повторюваними адміністративними завданнями, такими як реєстрація та оформлення квитків. У результаті відвідуваність збільшується, а витрати мінімізуються. Інструменти на місці включають друк значків, кіоски для реєстрації проїзду, онлайн-платежі та платформу квитків. У системі є додатки для Android та iOS, які перетворюють смартфони команди в точки доступу.

Веб-додаток Cvent також має розширені модулі для конкретних вимог, таких як управління готелями та подорожами, управління членством. Також передбачено управління бюджетом, звітування про події та маркетинг електронною поштою (рисунок 2.5).

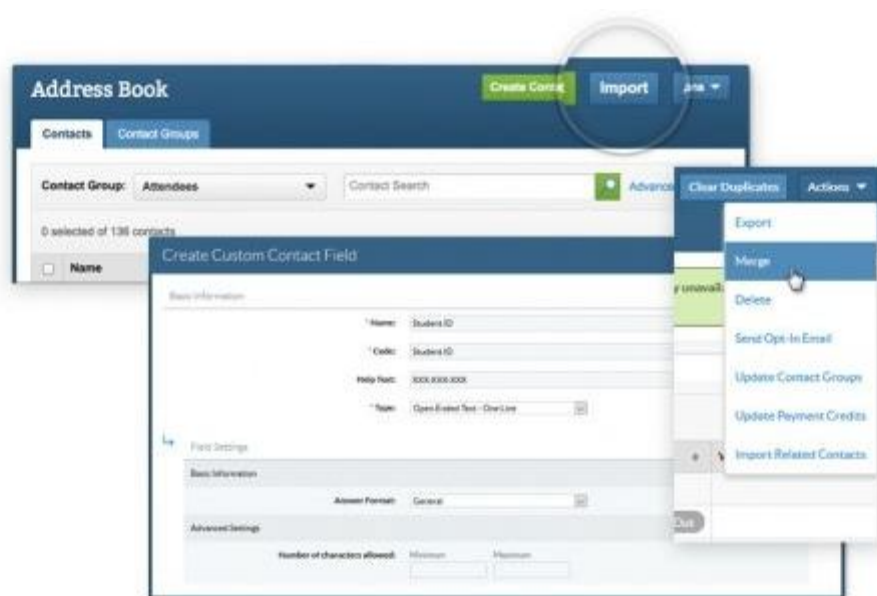


Рисунок 2.5 — Форми платформи Cvent для створення заходу [17]

Платформа Cvent Event Management надає менеджерам подій можливість створювати календарі подій, де відвідувачі можуть переглядати майбутні події, а також синхронізувати їх зі своїми календарями. Рішення надає можливості управління бюджетом, які відстежують кошти протягом кількох подій, розраховують кожен витрату на зустріч та загальні змінні витрати. Також Cvent Event Management має функцію звітування, яка пропонує вбудовані звіти, а також дає користувачам можливість створювати спеціалізовані звіти. Дані з усіх звітів також можна експортувати. Інші функції, пропоновані цим інструментом, включають користувацькі веб-сайти подій, управління контактами, веб-сайти та спільний доступ до соціальних мереж. Додаток Cvent пропонує інтеграцію з іншими сторонніми додатками, серед яких Salesforce, Concur, Marketo та Eloqua.

Застосунок Eventzilla [18] зручний щодо реєстрації та оформленні квитків на місці і заслуговує на високий рейтинг у списку програмного забезпечення 10 заходів для управління подіями за своєю чіткою інтуїтивно зрозумілою, простою функціональністю. Учасники можуть ввести свої дані в Інтернеті й завантажити електронні квитки зі свого Інтернет-порталу. Потім вони зареєструються через QR-код сканування QR-коду під час події. Ці функції значно скорочують чергування і забезпечують команді можливість зосередитися на проходах, логістиці, програмі та інших аспектах події.

Так само є дисконтні коди, які потребують більшої кількості замовлень. Також можна надсилати запрошення електронною поштою та отримувати сповіщення про оновлення через платформу. Якщо кількість учасників перевищить квоту, можна використовувати інструмент списку очікування для управління реєстрацією. Застосунок Eventzilla також інтегрується з високорівневими захищеними платіжними каналами PayPal, Stripe, Braintree і Authorize.net. Застосунок також є інструментом опитування після подій, який допоможе оцінити рівень успішності та постійно покращувати повторювані події. Додаток полегшує це за допомогою опитувань після подій, які можна налаштовувати за допомогою призначених для користувача питань, а також звітів з даними в реальному часі про трафік сторінок подій, продажах квитків і прибутки (рисунки 2.5).

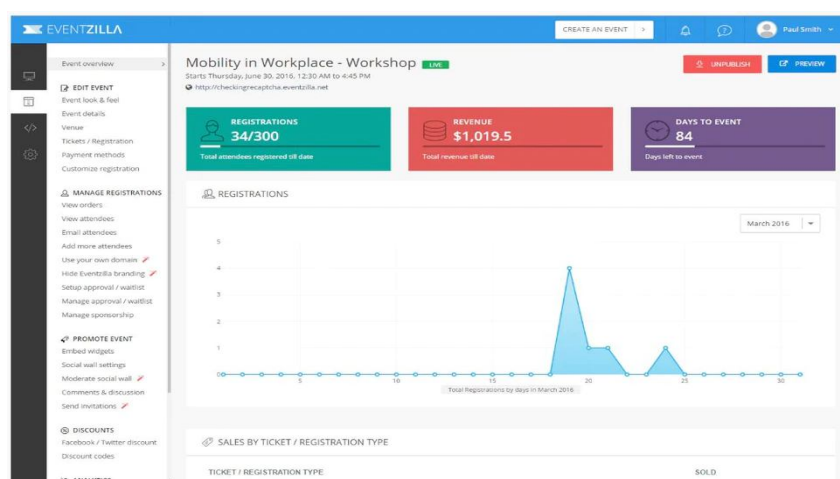


Рисунок 2.5 — Форми платформи Eventzilla для створення заходу [18]

Доступ до застосунку Eventzilla також можна отримати через мобільні додатки, доступні для пристроїв Android та iOS. Застосунок Eventzilla можна використовувати для фестивалів, конференцій, збору коштів, концертів, занять, мережеских заходів, спорту і багатьох інших подій. Платформа Eventzilla дає можливість менеджерам подій створювати спеціалізовані запитання щодо реєстрації подій для збору інформації про відвідувачів, таких як назва компанії, призначення і харчові уподобання. Користувачі також можуть створювати коди знижок і пропонувати спеціальні ціни для повторних клієнтів. Учасники можуть скористатися функцією мобільного реєстрації, яка дозволяє сканувати QR-коди. Застосунок пропонує інші функції, включаючи рекламу соціальних медіа, продаж квитків на місці, експорт списку відвідувачів, створення значків, управління приватними подіями та багато іншого; також пропонує базу знань, яка допомагає новим користувачам створювати і керувати подією.

3. ЗАСОБИ РОЗРОБКИ ПРОГРАМНОЇ ПЛАТФОРМИ

Правильно вибрані інструменти розробки є ключем до успіху будь-якого програмного забезпечення. Описані нижче засоби охоплюють широкий спектр потреб: від допомоги в організації роботи до завантаження і передачі файлів. Використовуючи всі можливості цих засобів розробки програмного забезпечення, дуже зручно працювати з проектами.

3.1 Мова програмування Typescript

Це мова з відкритим кодом, яка базується на JavaScript, одному з найбільш використовуваних інструментів у світі, додаючи статичні визначення типів. Типи — це спосіб описати форму об'єкта, надаючи кращу документацію і даючи можливість TypeScript перевірити правильність роботи коду. Типи написання можуть бути не обов'язковими для TypeScript, оскільки умова типу дозволяє отримати багато можливостей без написання додаткового коду [19]. Весь дійсний код JavaScript також є кодом TypeScript. Користувач може отримати повідомлення про помилки перевірки типу, але це не завадить запустити отриманий JavaScript.

Код TypeScript перетворюється в код JavaScript за допомогою компілятора TypeScript або Babel. Код JavaScript — чистий, простий код, який працює де завгодно: у браузері, на Node.JS або у програмах [20]. Зв'язок між TypeScript і JavaScript досить унікальний серед сучасних мов програмування. Код TypeScript розміщується як шар згори над JavaScript, пропонуючи функції JavaScript, а потім додає свій власний шар поверх них. Цей шар є системою типу TypeScript. У JavaScript вже є набір мовних примітивів, таких як рядок, число, об'єкт, невизначений, проте немає попередніх перевірок, які послідовно призначаються у всьому коді. Мова TypeScript виконує роль цього шару. Це означає, що діючий код

JavaScript також є кодом TypeScript, однак перевірка типу TypeScript може виділити розбіжності між тим, що розробник задав, що відбувається, і тим, що робить мова JavaScript.

Мова TypeScript перевершує інші аналоги такі, як мови програмування CoffeeScript і Dart, в тому сенсі, що TypeScript є розширенням JavaScript. Навпаки, такі мови, як Dart, CoffeeScript, є новими мовами і вимагають специфічної для мови середовища виконання.

Компіляція JavaScript — це мова, що інтерпретується. Отже, його потрібно запустити, щоб перевірити, що він дійсний. Транспортёр TypeScript надає функцію перевірки помилок. Мова TypeScript скомпілює код і згенерує повідомлення про помилки компіляції, якщо виявить якісь синтаксичні помилки. Це допомагає виділити помилки перед запуском скрипта.

Мова JavaScript не є строго типізованою. Вона поставляється з додатковою системою статичної типізації і виведення типів через TLS (служба мови TypeScript). Тип змінної, оголошеної без типу, може визначатися TLS на основі її значення. Мова TypeScript підтримує визначення типів для існуючих бібліотек JavaScript. Файл визначення TypeScript (з розширенням .d.ts) містить визначення для зовнішніх бібліотек JavaScript [21]. Отже, код TypeScript може містити ці бібліотеки. Мова TypeScript підтримує такі концепції об'єктно-орієнтованого програмування, як класи, інтерфейси, успадкування.

Основою мови TypeScript є компілятор TypeScript (tsc), який перетворює інструкції, написані мовою TypeScript, в їхній еквівалент мовою JavaScript.

Перевагами мови TypeScript порівняно з мовою JavaScript є те, що вона дає можливість явно визначати типи, підтримує використання повноцінних класів, як і в традиційних об'єктно-орієнтованих мовах, підтримує підключення модулів.

Мовна служба TypeScript надає додатковий шар навколо конвеєра основного компілятора, який є редактором додатків. Мовна служба підтримує загальний набір типових операцій редактора таких, як завершення операторів, довідка за підписом, форматування і виділення коду, розфарбування.

3.2 Бібліотека для створення інтерфейсу React

Бібліотека React — це бібліотека JavaScript, призначена для створення швидких й інтерактивних користувацьких інтерфейсів для веб-додатків і мобільних додатків [22]. Ця інтерфейсна бібліотека з відкритим початковим кодом, заснована на компонентах. Бібліотека обробляє тільки користувацький інтерфейс. Це відповідає видові у шаблоні MVC модель-вид-контролер (рисунок 3.1).

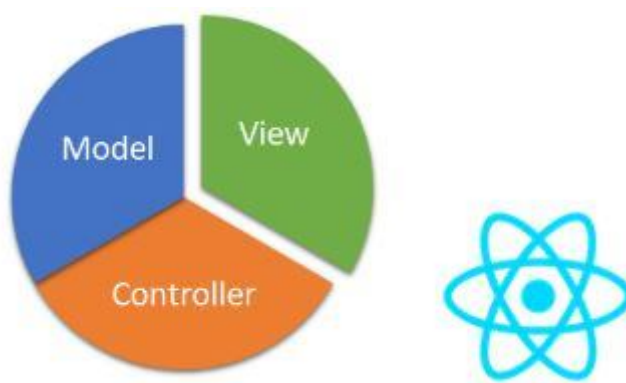


Рисунок 3.1 — Подання MVC-паттерну у вигляді діаграми

В основі всіх додатків React лежать компоненти. Компонент — це автономний модуль, який виводить деякий результат [23]. Використовуючи компоненти, можна написати елементи інтерфейсу, наприклад, такі як кнопка або поле введення. Компонент може включати в себе один або кілька інших компонентів. Взагалі кажучи, для створення програмного забезпечення треба написати компоненти React, які відповідають різним елементам інтерфейсу. Потім треба організувати ці компоненти всередині компонентів більш високого рівня, які визначають структуру застосування (рисунок 3.2). Наприклад, якщо взяти форму. Форма може складатися з безлічі елементів інтерфейсу, таких як поля введення, мітки або кнопки. Кожен елемент всередині форми може бути записаний як компонент React. Потім можна написати компонент більш високого рівня, сам компонент форми. Компонент форми визначатиме структуру форми і буде включати в себе кожен з цих елементів інтерфейсу.

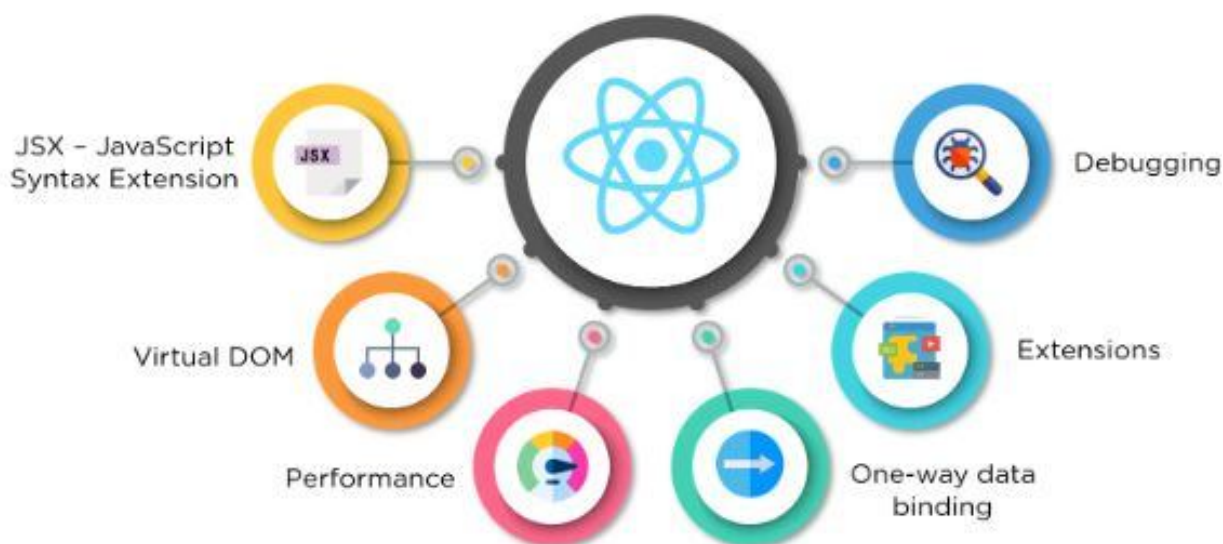


Рисунок 3.2 — Особливості React-фреймворку [24]

Важливо відзначити, що кожен компонент в додатку React дотримується суворих принципів управління даними [25]. Складні інтерактивні інтерфейси часто пов'язані зі складними даними і станом додатка. Поверхня React обмежена і націлена на те, щоб дати інструменти, які дадуть можливість передбачити, який вигляд буде мати додаток за таких обставин.

3.3 Мова запитів GraphQL

Запит мовою GraphQL — це рядок, який відправляється на сервер для інтерпретації й виконання; він потім повертає JSON назад клієнту. Визначає форму даних: запити GraphQL відображають їхню відповідь (рисунок 3.3). Це дає можливість легко передбачити форму даних, які повертаються із запиту, а також написати запит, маючи дані, які потрібні додатку. Мова GraphQL проста щодо освоєння і використання.

У мові GraphQL строго типізований кожен рівень запиту відповідає певному типу, і кожен тип описує набір доступних полів. Як і в SQL, це дає можливість GraphQL надавати описові повідомлення про помилки перед виконанням запиту. Кожне поле GraphQL на сервері підтримується будь-якою довільною функцією.



Рисунок 3.3 — Приклад GraphQL-запиту

Мова GraphQL не забезпечує ніякого резервного сховища [26]. Замість цього GraphQL використовує переваги існуючого коду. Форма даних, які повертаються, повністю визначається запитом клієнта, тому сервери спрощуються і легко узагальнюються.

Ця мова гарантує, що додаток завантажує тільки релевантні і абсолютно необхідні дані, навіть якщо вони з декількох джерел. Хоча GraphQL може видатися незвичайною новою технологією, створеної Facebook, багато хто вважає, що реалізація насправді дуже практична.

Мова GraphQL зарекомендувала себе як рішення для збору даних з кількох джерел, визначення даних і опису даних [27]. Якщо розробник, який використовує дану технологію, зрозуміє деякі з базових концепцій, які можуть отримати вигоду з GraphQL, може в кінцевому підсумку призвести до того, що він стане основою для майбутнього розвитку.

3.4 База даних MongoDB

База даних MongoDB — це база даних з відкритим вихідним кодом, в якій використовується модель даних, орієнтована на документи, і неструктурована мова

запитів. На сьогоднішній день це одна з найпотужніших систем і баз даних NoSQL [28]. Мова NoSQL означає, що вона не використовує звичайні рядки і колонки, які користувачі пов'язують з керуванням реляційними базами даних. Це архітектура, яка заснована на колекціях і документах (рисунок 3.4). Основна одиниця даних в цій базі даних складається з набору пар ключ-значення [29]. Це дає можливість документам мати різні поля і структури.

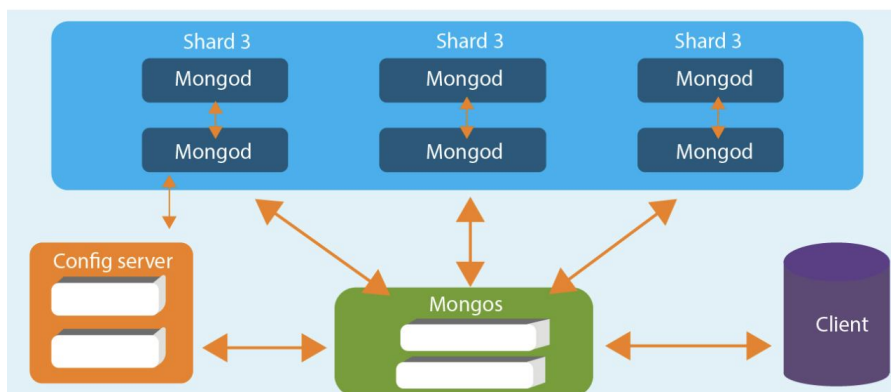


Рисунок 3.4 — Архітектура бази даних NoSQL [30]

У цій базі даних використовується формат зберігання документів BSON, що представляє собою двійковий стиль JSON.документи.

Модель даних MongoDB дає можливість комбінувати і зберігати дані багатовимірних типів без шкоди для потужних опцій індексування, доступу до даних і правил перевірки [31]. Там немає затримки, якщо потрібно динамічно змінювати схеми.

Переваги MongoDB:

— кожна база даних містить колекції, які в свою чергу містять документи. Кожен документ може бути різним з різною кількістю полів. Розмір і зміст кожного документа можуть відрізнятися один від одного;

— структура документа більше відповідає тому, як розробники конструюють свої класи і об'єкти відповідними мовами програмування. Розробники часто говорять, що їхні класи не є рядками і колонками, а мають чітку структуру з парами ключ-значення;

— рядки (або документи, які викликаються в MongoDB) не обов’язково повинні мати наперед визначену схему. Замість цього поля можуть бути створені на льоту;

— модель даних, доступна в MongoDB, дає можливість представляти ієрархічні відношення, простіше зберігати масиви та інші більш складні структури.

3.5 Сервіс Cloudinary

Сервіс Cloudinary — це сервіс, який управляє додатками, пов’язаними з зображеннями і відео, в хмарі. Він пропонує широкий спектр інструментів управління мультимедіа, що використовуються багатьма розробниками веб-додатків і мобільних додатків по всьому світу. Завдяки своїй назві Cloudinary дає користувачам можливість легко завантажувати і зберігати зображення і відео в хмарі. Користувачам також більше не потрібні окремі програми для зміни розміру, обрізки, перетворення і виконання інших подібних дій, оскільки сервіс пропонує широкі можливості редагування.

Сервіс Cloudinary також може бути легко інтегрований з будь-яким веб-додатком з його великими можливостями Application Program Interface (API). Його надійні мережі доставки контенту (CDN) також забезпечують швидку доставку контенту. Компанії можуть покінчити з дорогими і складними інструментами управління графікою, оскільки Cloudinary представляє комплексні рішення для мультимедійних потреб будь-якої організації.

Сервіс підтримує завантаження зображень, маніпуляції і оптимізацію для Інтернету, хмарного сховища і доставки. Він надає API і гнучкі функції адміністратора для інтеграції з існуючими та новими мобільними і веб-додатками.

Цей сервіс дає можливість завантажувати і зберігати необмежену кількість зображень безпечно і в приватному порядку. Сховище включає в себе історичні ревізії і автоматичне резервне копіювання. Користувач може маніпулювати зображеннями, використовуючи різні варіанти, такі як водяні знаки, розпізнавання осіб, кадрування, зміна розміру, застосування ефектів і багато іншого.

Зображення доставляються через всесвітній CDN Akamai для швидкої доставки, оптимізованої для всіх пристроїв. Сервіс Cloudinary, таким чином, являє собою платформу управління цифровими активами з бібліотекою мультимедіа, яка використовує SDK та API-інтерфейси RESTful для автоматизації управління зображеннями. Нарешті, можливо використовувати звіти і розширену аналітику для розуміння і оптимізації продуктивності зображень.

3.6 Редактор коду Microsoft Visual Code

За свою довгу історію існування Microsoft випустив чимало інструментів розробки. Але так уже склалося, що тільки Visual Studio — велике і потужне IDE, призначене для всього. Розвивається цей продукт вже більше двох десятиріч і має в собі різноманітні функції. Багатьом цей інструментарій подобається і іноді навіть задавали питання — чи буде перенесений Visual Studio на інші платформи. Можливо, таке портування буде дорогим і невиправдано складним, оскільки дуже багато всього в цьому IDE зав'язано на Windows.

У результаті Microsoft представив новий продукт під назвою Visual Studio Code, який працює відразу на трьох платформах: Linux, Mac OS і Windows. Середовище Visual Studio Code — це саме редактор, але при цьому має функції IDE, що покладається на розширення.

Як і багато редакторів коду, VS Code використовує поширену схему розташування основних елементів — зліва оглядач файлів, праворуч редактор коду. На додаток до цього є специфічні елементи, які забезпечують навігацію і виконання спеціальних команд.

4. РОЗРОБКА ПЛАТФОРМИ

Створено веб-платформу для організації міжнародних конференцій з використанням 3 головних модулів: клієнтська частина, серверна частина і сервер баз даних (рисунок 4.1).

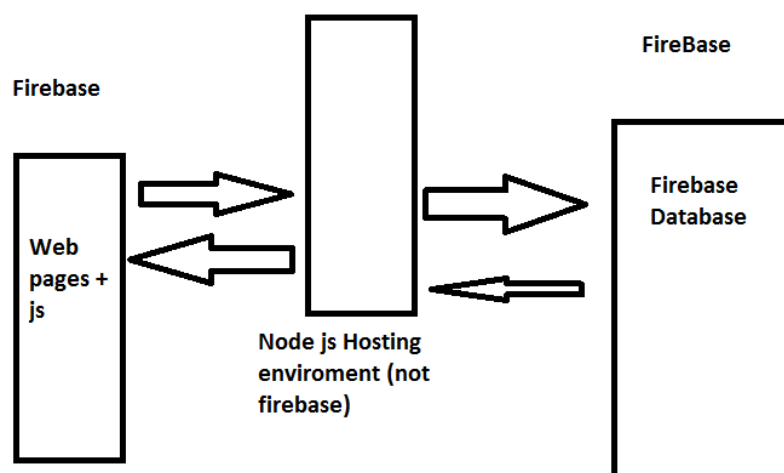


Рисунок 4.1 — Діаграма архітектури веб-платформи

Така доступна і проста архітектура дає можливість дуже просто підтримувати і масштабувати контейнери для розгортання production-версії.

4.1 Архітектура веб-платформи

Платформа створена на основі клієнт-серверного шаблону архітектури веб-додатків (рисунок 4.2). Загальна концепція архітектури даної платформи відповідає концепції користувача браузера, який запускає додаток, що може працювати на декількох веб-сайтах [23]. В основі розробки серверної частини було використано Docker для безпечного й ізольованого запуску додатку і подальшого розгортання в production. Для початку роботи користувачеві потрібно запустити клієнтську і серверну частини. У системі є такі моделі: User (користувач), Conference (дані про

конференцію), Ticket (Дані про квитки). Основною сутністю платформи є Conference, учасниками або організаторами є користувачі.

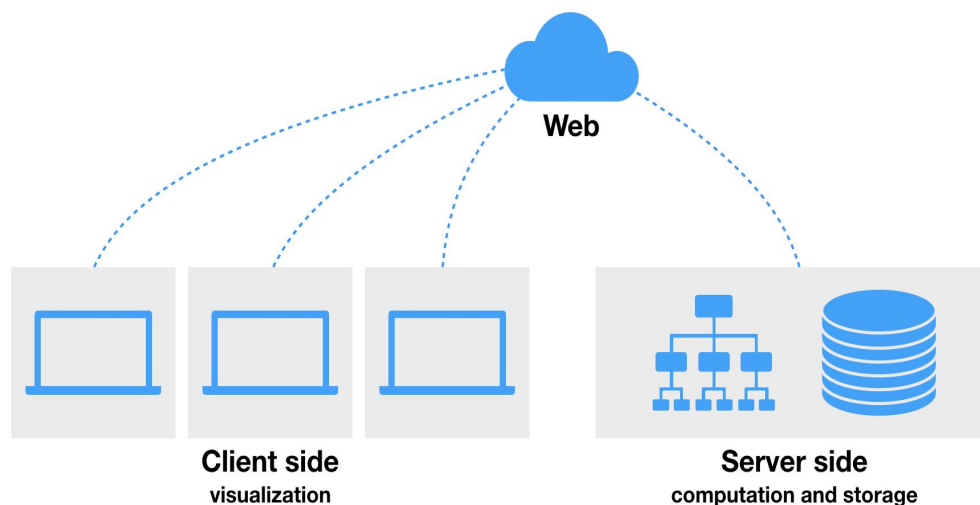


Рисунок 4.2 — Клієнт-серверна архітектура [32]

Основою процесу розробки додатку є такі структурні компоненти: веб-браузер або клієнт; сервер веб-додатку; сервер бази даних.

На діаграмах (рисунки 4.3-4.4) подано загальний вигляд системи, дії користувача, та описані моделі кожної сутності.

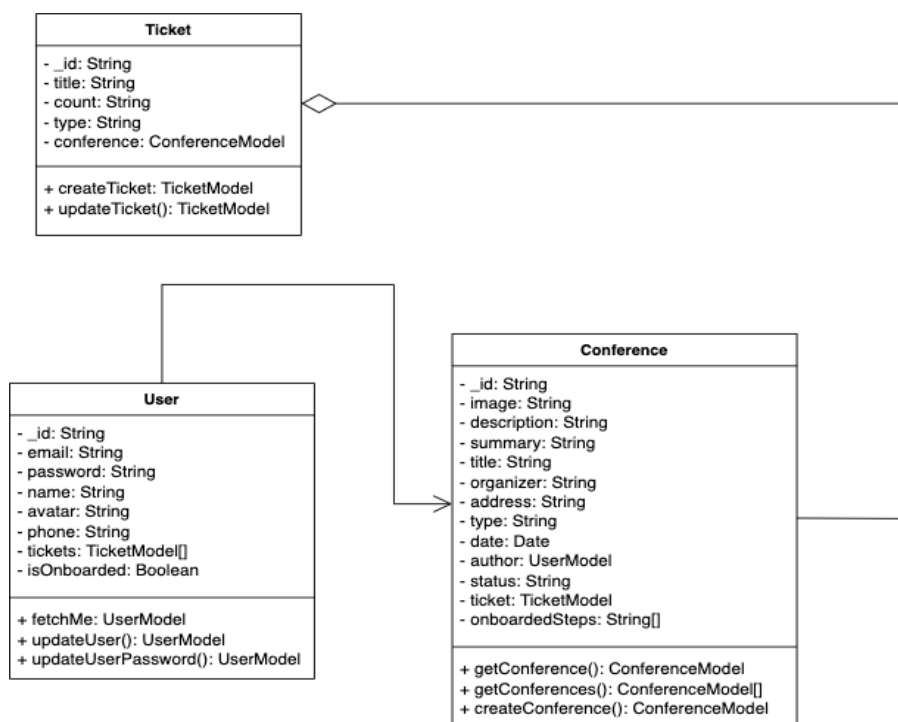


Рисунок 4.3 — Діаграма класів



Рисунок 4.4 — Діаграма прецедентів

Така архітектура веб-додатку відрізняється не тільки за ефективністю, але також за надійністю, масштабованістю, безпекою і надійністю.

4.2 Створення сторінки перегляду конференцій користувачем

Сторінка перегляду конференцій користувача створена для відображення списку конференцій у вигляді таблиці, в яких користувач є створювачем або учасником. Сторінка розміщується за адресою /conferences, має посилання на сторінку для створення конференцій. Кожен елемент таблиці (конференція як сутність) відображає:

- назву конференції (title),
- адресу (address);
- час початку проведення (startDate);
- головне зображення (image);
- статус (status);
- меню з випадаючим меню з кнопками edit, publish та preview.

Також реалізовано перенаправлення по кліку миші на сторінку детального перегляду конференції. Це дає користувачам можливість здійснювати зручну і просту навігацію по таблиці.

4.3 Розробка сторінки для створення нової конференції

Сторінка для створення нової конференції представлена у вигляді форми з полями вводу, випадаючими меню та меню для вибору дати.

Алгоритм створення нової конференції подано на рисунку 4.5.



Рисунок 4.5 — Алгоритм створення конференції

Сторінка розміщується за адресою `/conferences/create`, має кнопку для відправлення ведених даних на сервер.

Головна форма створення ділиться на 3 підформи: основна інформація, місце проведення, дата і час. Підформа для основної інформації має поля: Назва конференції, Коротка інформація про організатора.

Підформа для місця проведення в свою чергу має поля: адреса проведення або посилання на онлайн-презентацію при обраному типі місця проведення — онлайн-конференція.

Підформи Дата і Час мають поля для вибору дати початку і закінчення конференції. Після створення в базі даних нової конференції сервер повертає відповідь на клієнт, який обробляє відповідь і відображає нову конференцію або, в разі помилки, повертається тіло помилки для відображення відповідного сповіщення.

Підформи також об'єднанні у загальну форму для читабельності та простого розуміння коду. Форма для створення конференції має валідацію для обов'язкових полів. Введені користувачем дані відправляються на сервер для створення нового документа в базі даних.

4.4 Розробка сторінки для оновлення інформації

Сторінка для оновлення детальної інформації створена у вигляді головної форми, яка в свою чергу ділиться на три підформи: головне тематичне зображення (блок для скидання або вибору файлів), файли конференції (кнопка для вибору файлів) та опис.

Алгоритм оновлення інформації про конференцію подано на рисунку 4.6.

Сторінку для оновлення інформації можна знайти за адресою `/conferences/ідентифікатор_конференції/details`.

На сторінці є кнопка для оновлення введених даних в базі даних. Підформа для опису має: поле для введення короткого підсумку, а також має спеціальний редактор для створення форматованих даних.

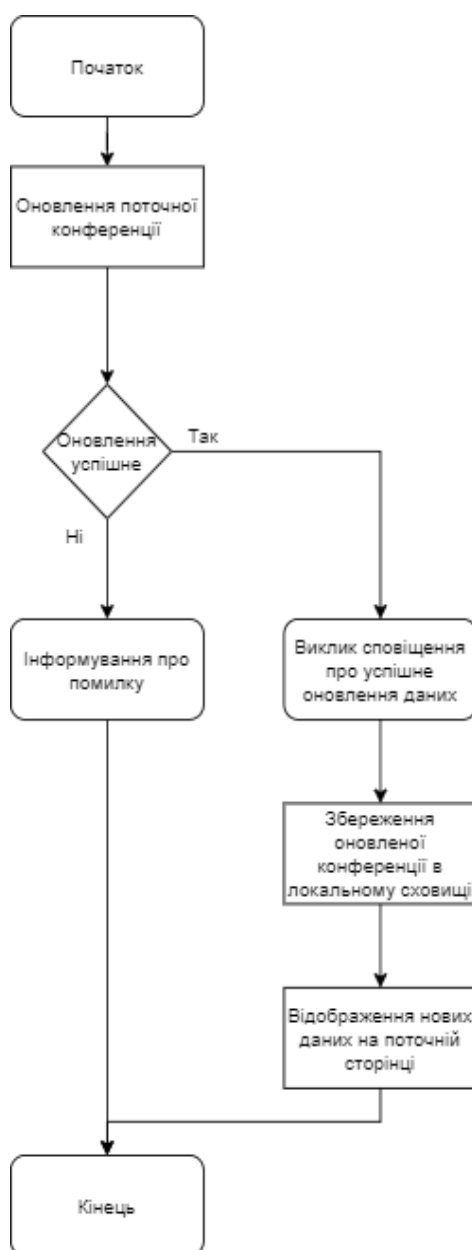


Рисунок 4.6 — Алгоритм оновлення інформації про конференцію

Для зберігання завантажених зображень використовується хмарне сховище Cloudinary, яке дає безкоштовний доступ на 25Gb пам'яті кожні 30 днів.

Після успішного завантаження файлів на Cloudinary цей сервіс повертає дані, в яких є поле з посиланням на файл (рисунок 4.7). Таким чином це посилання повертається на клієнт та відображається у відповідній формі.

Тому при натисканні на кнопку Update, всі введені дані у формі детальної інформації разом з посиланнями на завантажені файли відправляються на сервер і в свою чергу зберігаються.



Рисунок 4.7 — Алгоритм зберігання і видалення файлів

На схемі, поданій вище (рисунок 4.6) продемонстровано просту і безпечну логіку для пошуку і оновлення даних в конференції. За допомогою такої вибірки тільки автор конференції може змінити дані, який авторизований в системі. Також у випадку помилки, користувач отримає відповідне сповіщення.

4.5 Розробка сторінки квитків на конференцію

Головна сторінка для роботи з квитками складається з 2 головних компонентів: таблиця для відображення наборів квитків і модальне вікно для створення, редагування і видалення набору квитків. Сторінка розміщується за адресою `/conferences/ідентифікатор_конференції/tickets`, має кнопку для відкриття модального вікна з формою для створення нового набору квитків. Кожен елемент у

таблиці відображає: кількість квитків, тип квитків. Випадаюче меню з кнопками для виклику модального вікна з заповненими поточними даними.

Розроблена веб-платформа допоможе спростити планування конференцій і надасть інструменти, необхідні для створення й подальшого супроводу таких заходів.

Використовуючи розроблену веб-платформу, планувальники подій можуть автоматизувати всі функції програмного забезпечення для управління подіями.

5. ОПИС РОБОТИ КОРИСТУВАЧА З ПЛАТФОРМОЮ

Для використання функціоналу програмною платформою було розроблено простий і зручний інтерфейсу. Основний користувацький інтерфейс було розроблено з використанням бібліотеки Antd.

5.1 Системні вимоги

Для початку роботи з програмою потрібно встановити додаткове програмне забезпечення: NodeJS, Docker та MongoDB безпосередньо на пристрої. Після встановлення додаткових програм для запуску додатку треба виконати команди, прописані у файлі README.md. У результаті виконання команд на пристрої запуститься серверна частина (Back End) і клієнтська частина (Front End).

Для запуску і роботи системи необхідні такі умови: вільне місце на жорсткому диску ~ 45 Гб, оперативна пам'ять (RAM) 2 Гб.

5.2 Використання платформи

Перед початком роботи з платформою користувач має можливість обрати локалізацію мови, з якою він буде працювати в подальшому. Локалізація представлена в двох основних мовах — англійська та українська (рисунок 5.1).

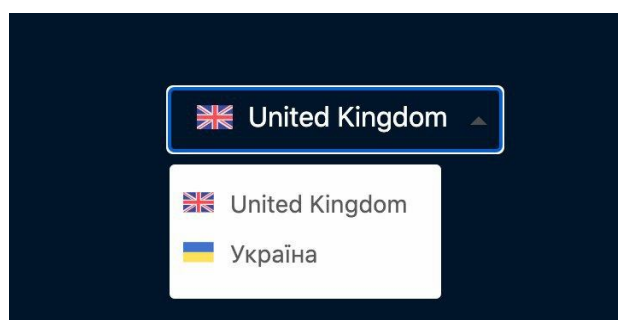


Рисунок 5.1 — Випадаючий список для вибору мови

Функціонал локалізації також має автоматичний вибір мови користувача. Можливості використання розробленої програмної системи відображено на рисунках 5.2-5.7.

Рисунок 5.2 — Сторінка для входу в систему

Рисунок 5.3 — Сторінка для реєстрації нового користувача

Після реєстрації або вводу логіну, система перенаправляє користувача до його особистого кабінету, де він може створити першу конференцію, якщо вона ще не була створена.

На рисунку 5.5 Location відповідає за тип локації проведення. Якщо обрати тип Venute, то потрібно ввести реальну адресу місця проведення конференції, але якщо обрано тип Online Conference, то замість поля для введення адреси з'явиться поле для прикріплення посилання на онлайн-конференцію, презентацію або відео в режимі реального часу.

У підформі Date and time, є два види кількостей проведення конференції. За замовчуванням обрано тип для проведення одноразової конференції за визначеним часом. Якщо вибрати тип повторного проведення конференції, то з'явиться спеціальний CalendarPicker для вибору будь-яких дат проведення конференцій. Після натискання на кнопку для створення нової конференції, система переадресує користувача, у разі успіху, на сторінку для введення детальних даних (рисунки 5.6-5.7).

Sarpo Conferences Preview Publish Conference [User Profile]

Basic Info

Name your event and tell event-goers why they should come. Add details that highlight what makes it unique.

Enter conference name

Tell attendees who is organizing this event

Location

Help people in the area discover your event and let attendees know where to show up.

☒ Venue ☐ Online conference

Enter address

Date and time

Tell event-goers when your event starts and ends so they can make plans to attend

Frequency

☒ Single Event - Happens once and can last multiple days ☐ Recurring Events - Repeats or occurs more than once

Start and end times

Start date → End date

Start time → End time

Рисунок 5.4 — Сторінка програмної платформи для організації міжнародних конференцій

Sarpo Conferences Preview Publish Conference [User Profile]

Basic Info

Name your event and tell event-goers why they should come. Add details that highlight what makes it unique.

Enter conference name

Tell attendees who is organizing this event

Location

Help people in the area discover your event and let attendees know where to show up.

☒ Venue ☐ Online conference

Enter address

Date and time

Tell event-goers when your event starts and ends so they can make plans to attend

Frequency

☒ Single Event - Happens once and can last multiple days ☐ Recurring Events - Repeats or occurs more than once

Start and end times

Start date → End date

Start time → End time

Create

Рисунок 5.5 — Форма для створення нової конференції

На рисунках 5.6 і 5.7 подано вигляд форми для введення детального опису конференції. Для відображення головного зображення конференції в списку конференції або безпосередньо на сторінці перегляду використовується форма для завантаження тематичного зображення конференції. На сторінці перегляду також будуть відображені у вигляді списка прикріплені через форму Files, файли, які при натисканні на назву файлу, автоматично будуть завантажені.

The screenshot shows the 'Test ConferenceS' management page. On the left, a sidebar contains links for 'Basic Info', 'Details' (which is highlighted), 'Tickets', and 'Dashboard'. The main content area is titled 'Main Conference Image' and includes the instruction 'This is the first image attendees will see at the top of your listing.' Below this is a large dashed rectangular box for image upload, with a camera icon and the text 'Click or drag file to this area to upload'. A note below the box states: 'Support for a single or bulk upload. Strictly prohibit from uploading company data or other band files'. Below the image section is the 'Files' section, which says 'Upload files to display them on preview page'. It lists two files: 'document.pdf' and 'document (1).pdf', each with a trash icon to its right. An 'Upload' button is at the bottom of this section. The 'Description' section follows, with the prompt 'Write a short event summary to get attendees excited'. It contains a text input field with the placeholder 'Summary'.

Рисунок 5.6 — Сторінка для введення детального опису конференції

This screenshot shows the 'Description' section of the same interface. It includes the same sidebar and the 'Files' section above. The 'Description' section has a text input field for a 'Summary'. Below it is a larger text area containing a numbered list: '1. Test', '2. Test', and '3. Test'. At the bottom of this text area is a rich text editor toolbar with icons for bold (B), italic (I), underline (U), strikethrough (ABC), bulleted list, numbered list, and link. At the very bottom of the form is a blue 'Update' button.

Рисунок 5.7 — Продовження сторінки для введення детального опису конференції

На рисунку 5.8 подано приклад створеного набору квитків. При натисканні на кнопку Create Ticket (Створити квиток) буде негайно викликано модальне вікно з формою для створення квитка.

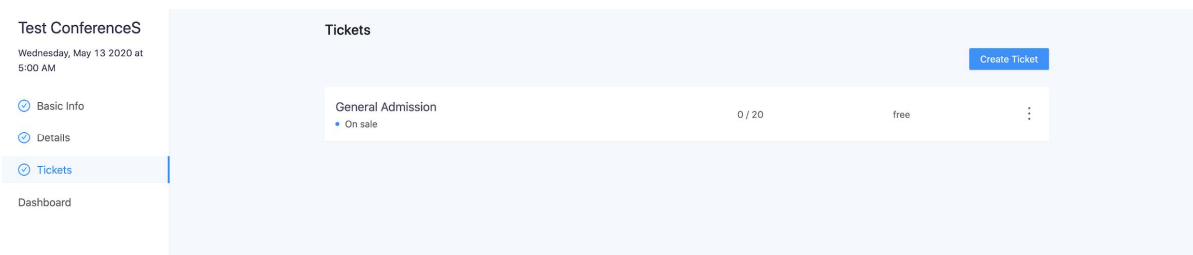


Рисунок 5.8 — Сторінка для створення та перегляду квитків

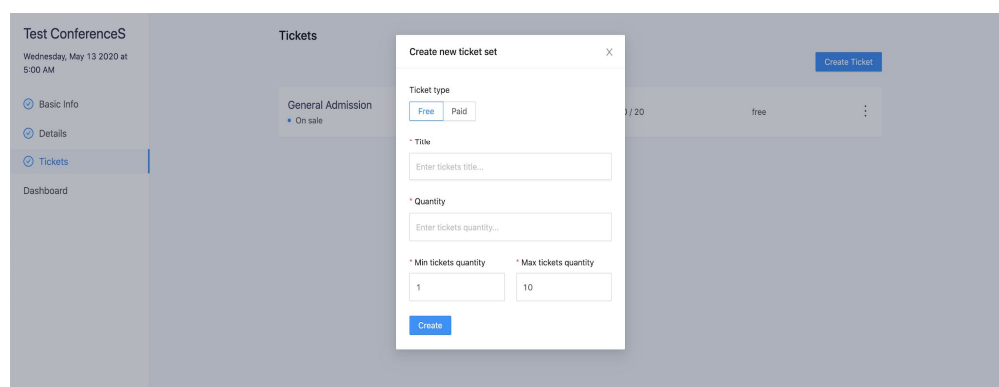


Рисунок 5.9 — Модальне вікно з формою для створення квитків

Форма модального вікна для створення квитків передбачає вибір типу для кожного квитка. Тип квитка може бути платним або ж безкоштовним. Якщо обраний платний тип, серед полів форми з'явиться додаткова поле для введення ціни за кожен квиток.

ВИСНОВКИ

При написанні дипломної роботи було проведено аналіз предметної області і вимог, розроблено архітектуру і створено веб-платформу для організації міжнародних конференцій. Створене програмне забезпечення призначене як організаторів, так і для учасників конференцій.

У розробленій платформі враховано функціонал існуючих веб-додатків, складовими якого є авторизація, реєстрація, редагування даних про користувача, можливість збирати різноманітні матеріали, переглядати матеріали поточної і попередніх конференцій. Також є можливість організатору створювати будь-яку кількість запрошень, виконати їхнє редагування, видалення. Користувач при створенні або редагуванні заходу може прикріпити файли різного типу, зображення, опис, карту місцезнаходження, вказати час і дату проведення заходу. На головній сторінці веб-платформи виконується фільтрація за датою проведення заходу.

Створений веб-додаток має три компоненти: головна сторінка для перегляду конференцій, головна панель для їхнього створення і модуль для редагування персональних даних користувачів. При розробці модулів веб-платформи для написання клієнтської частини вибрано мову Typescript і веб-фреймворк ReactJS, для серверної — NodeJS, також використано мову запитів і маніпулювання даними GraphQL, базу даних MongoDB. Розроблена веб-платформа допоможе спростити планування конференцій і надасть інструменти, необхідні для створення й подальшого супроводу таких заходів.

Результати роботи доповідалися на конференції “Сучасні проблеми наукового забезпечення енергетики” [31].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cappel L. Event management for dummies / Laura Capell // John Wiley & Sons, Ltd, 2013 — P. 17.
2. Allen J. Event Planning Ethics and Etiquette: A Principled Approach to the Business of Special Event Management / Juddy Allen // Wiley, 1-st edition, July 30, 2009. — P. 261-263.
3. Paulette W. Event Planning Made Easy / Paulette Wolf // McGraw-Hill Education, 1-st edition, May, 11, 2005. — P. 103-104.
4. Dowson R. Event Planning and Management: Principles, Planning and Practice / Ruth Dowson // Kogan Page, 2-nd edition, November, 28, 2018/ — P. 351.
5. Goldblatt J.— Special Events: A New Generation and the Next Frontier / Joe Goldblatt // Wiley; 6-th edition, October, 5, 2010. — P. 474-475.
6. What Is Event Management Software? Analysis of Features, Pricing, Types and Benefits [Електронний ресурс] — Режим доступу: <https://financesonline.com/event-management-software-analysis-features-pricing-types-benefits>.
7. Event Management Market and Event Management Software Market [Електронний ресурс] — Режим доступу: <https://www.globenewswire.com/news-release/2020/02/18/1985967/0/en/Event-Management-Market-and-Event-Management-Software-Market-An-Analysis-of-the-Global-Key-Players-Technical-Evolution-New-Marketing-Paradigm-Product-Development-Growth-Overview-Re.html>
8. Event management software [Електронний ресурс] — Режим доступу: <https://www.predictiveanalyticstoday.com/top-event-management-software/>
9. Allen J. Event Planning: The Ultimate Guide To Successful Events / Judy Allen // Wiley; 2-nd edition, December, 30, 2008. — P. 128-129.

10. Global Event Management Software Market [Електронний ресурс]. — Режим доступу: <https://www.marketresearchfuture.com/reports/event-management-software-market-1399>
11. Meegan J. Sustainable Event Management: A Practical Guide / Meegan Jones // Earthscan Publications Ltd.; 1-st editio, January, 2010. — P. 175.
12. Allison S. The Event Marketing Handbook: Beyond Logistics & Planning / Allison Saget // CreateSpace Independent Publishing Platform; Edition Unstated edition, August, 11, 2012. — P. 48.
13. Сервіс для організації заходів Monday [Електронний ресурс]. — Режим доступу: <https://monday.com>
14. Cherny B. Programming TypeScript: Making Your JavaScript Applications Scale / Boris Cherny // O'Reilly Media, 1-st edition, May, 23, 2019. — P. 21-22.
15. Сервіс для організації заходів EventBookings [Електронний ресурс]. — Режим доступу: <https://www.eventbookings.com>
16. Сервіс для організації заходів Gather [Електронний ресурс]. — Режим доступу: <https://www.gatherhere.com>
17. Сервіс для організації заходів Cvent [Електронний ресурс]. — Режим доступу: <https://www.cvent.com>
18. Сервіс для організації заходів Eventzilla [Електронний ресурс]. — Режим доступу: <https://www.capterra.com/p/117828/Eventzilla>
19. Freeman A. Essential TypeScript: From Beginner to Pro / Adam Freeman // Apress; 1st edition, August, 15, 2019. — P. 112.
20. Basarat A. TypeScript Deep Dive / Basarat Ali Syed // Samurai Media Limited, June, 6, 2017. — P. 36.
21. Banks A. Learning React: Functional Web Development with React and Redux / Alex Banks // O'Reilly Media; 1-st edition, May, 18, 2017. — P. 127-128.
22. Wieruch R. The Road to React: Your journey to master plain yet pragmatic React.js / Robin Wieruch // O'Reilly Media; 1-st edition, August, 9, 2016. — P. 45.
23. Stefanov S. React: Up & Running / Stoyan Stefanov // O'Reilly Media; 1-st edition, August, 9, 2016. — P. 74-76.

24. What is React? [Електронний ресурс]/ — Режим доступу: <https://www.simplilearn.com/what-is-react-article>
25. Porcello E. Learning GraphQL: Declarative Data Fetching for Modern Web Apps / Eve Porcello // O'Reilly Media; 1-st edition, September, 11, 2018. — P. 112.
26. Wieruch R. The Road to GraphQL: Your journey to master pragmatic GraphQL in JavaScript with React.js and Node.js / Robin Wieruch // O'Reilly Media; 1-st edition, September, 11, 2018. — P. 85-86.
27. Chodorow K. MongoDB: The Definitive Guide / Kristina Chodorow // O'Reilly Media; 2-nd edition, June, 4, 2013. — P. 254-255.
28. Banker K. MongoDB in Action / Kyle Banker // Manning Publications; 1-st edition, December, 27, 2011. — P. 182-183.
29. Membrey P. The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing / Peter Membrey // Apress; 1st ed., October, 26, 2010. — P. 328-329.
30. Shklar L. Web Application Architecture: Principles, Protocols and Practices / Leon Shklar // Wiley; 2-nd edition, April, 27, 2009. — P. 432-433.
31. Гладкий О.Л. Веб-платформа для організації міжнародних конференцій / О.Л. Гладкий, Л.І. Кублій // Сучасні проблеми наукового забезпечення енергетики: Матеріали XVIII Міжнародної науково-практичної конференції молодих вчених і студентів 2020 року. — К.: КПІ ім. Ігоря Сікорського, 2020. — Т. 2. — С. 118.

ДОДАТОК А

Програмна платформа для організації міжнародних конференцій

Специфікація

УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_20Б

Аркушів 2

Київ 2020

| Позначення | Найменування | Примітки |
|--|--------------|-------------------------------------|
| Документація | | |
| УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_2 0Б | Записка.docx | Текстова частина дипломної роботи |
| Компоненти | | |
| УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_2 0Б 12-1 | Sarpi.api | Основний модуль серверної частини |
| УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_2 0Б 12-2 | Sarpo.web | Основний модуль клієнтської частини |

ДОДАТОК Б

Програмна платформа для організації міжнародних конференцій

Текст програми

УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_20Б

Аркушів 2

Київ 2020

```

import React, { useEffect } from 'react';

import { useDispatch, useSelector } from 'react-redux';

import ConferencesPageView from '../components/ConferencesPageView';

import { fetchConferences } from '../ConferenceActions';

import { getConferences } from '../ConferenceReducer';

const ConferencesPage: React.FunctionComponent = () => {

  const conferences = useSelector(getConferences);

  const dispatch = useDispatch();

  useEffect(() => {

    dispatch(fetchConferences());

  }, []);

  return <ConferencesPageView conferences={conferences} />;

};

export default ConferencesPage;

import { Button } from 'antd';
import React from 'react';
import { Link } from 'react-router-dom';
import { useTranslation } from 'react-i18next';
import styled from 'styled-components';

import LayoutHeader from '@components/LayoutHeader';
import Footer from '@components/Footer';

import ConferencesTable from './ConferencesTable';

interface ConferencesPageViewProps {
  conferences: any[];
}

const ConferencesPageView: React.FunctionComponent<ConferencesPageViewProps> = ({
  conferences,
}: ConferencesPageViewProps) => {
  const { t } = useTranslation();

```

```

return (
  <Wrapper>
    <LayoutHeader />
    <Content>
      <TableHeader>
        <Title>{t('Conferences')}</Title>
        <Link to="/conferences/create">
          <Button type="primary">{t('Create conference')}</Button>
        </Link>
      </TableHeader>
      <ConferencesTable conferences={conferences} />
    </Content>
    <Footer />
  </Wrapper>
);
};

```

```
const Wrapper = styled.div``;
```

```

const Content = styled.div`
  min-height: calc(100vh - 64px);
  margin: 0 10%;
  padding-top: 50px;
`;

```

```

const TableHeader = styled.div`
  display: flex;
  justify-content: space-between;
`;

```

```

const Title = styled.div`
  color: #1e0a3c;
  font-size: 1.875rem;
  font-weight: 800;
`;

```

```
export default ConferencesPageView;
```

```

import { Table } from 'antd';
import React from 'react';
import styled from 'styled-components';

```

```
import { TitleRender } from './ConferencesTableColumns';
```

```

const columns = [
  {
    title: 'Conference',
    dataIndex: 'conference',
    key: 'conference',

```

```

    // eslint-disable-next-line react/display-name
    render: (_, record: Record<string, any>): any => (
      <TitleRender record={record} />
    ),
  },
  {
    title: 'Sold',
    dataIndex: 'sold',
    key: 'sold',
    // eslint-disable-next-line react/display-name
    render: (): any => <div>0 / 0</div>,
  },
  {
    title: 'Status',
    dataIndex: 'status',
    key: 'status',
    // eslint-disable-next-line react/display-name
    render: (): any => <div>draft</div>,
  },
];

interface ConferencesTableProps {
  conferences: any[];
}

const ConferencesTable: React.FunctionComponent<ConferencesTableProps> = ({
  conferences,
}: ConferencesTableProps) => (
  <Wrapper>
    <Table
      columns={columns}
      dataSource={conferences}
      rowKey={(conference: Record<string, any>): string => conference._id}
    />
  </Wrapper>
);

const Wrapper = styled.div`
  margin-top: 25px;
`;

export default ConferencesTable;
import { Form } from 'antd';
import React from 'react';
import { useDispatch } from 'react-redux';

import { normalizeBasicConferenceToCreate } from '../utils/conference-helpers';
import { createConference } from '../ConferenceActions';
import CreateConferencePageView from '../components/CreateConferencePageView';

```



```

const CreateConferencePage: React.FunctionComponent = () => {
  const [form] = Form.useForm();

  const dispatch = useDispatch();

  const onFinish = async (values: Record<string, any>): Promise<void> => {
    const normalizedData = normalizeBasicConferenceToCreate(values);

    await dispatch(createConference(normalizedData));
  };

  return <CreateConferencePageView form={form} onFinish={onFinish} />;
};

export default CreateConferencePage;

import { FormInstance } from 'antd/lib/form';
import React from 'react';
import styled from 'styled-components';

import LayoutHeader from '@components/LayoutHeader';
import Footer from '@components/Footer';

import CreateOrEditConferenceForm from '../forms/ConferenceCreateOrEditForm';

interface CreateConferencePageViewProps {
  form: FormInstance;
  onFinish: (values: Record<string, any>) => Promise<void>;
}

const CreateConferencePageView: React.FunctionComponent<CreateConferencePageViewProps> = ({
  form,
  onFinish,
}: CreateConferencePageViewProps) => (
  <Wrapper>
    <LayoutHeader />
    <FormWrapper>
      <CreateOrEditConferenceForm form={form} onFinish={onFinish} />
    </FormWrapper>
    <Footer />
  </Wrapper>
);

const Wrapper = styled.div`
  margin-bottom: 40px;
  height: 100vh;
`;

const FormWrapper = styled.div`
  margin: 0 25% 30px 25%;

```

```

input {
  height: 45px;
}

@media (max-width: 768px) {
  margin: 0 40%;
}

@media (max-width: 1024px) {
  margin: 0 15%;
}
`;

export default CreateConferencePageView;

import { Button, Form } from 'antd';
import { FormInstance } from 'antd/lib/form';
import React from 'react';
import styled from 'styled-components';

import ConferenceBasicInfoForm from './ConferenceBasicInfoForm';
import ConferenceLocationForm from './ConferenceLocationForm';
import ConferenceDateForm from './ConferenceDateForm';
import { ConferenceType, ConferenceDateType } from '../models/conference';

interface ConferenceCreateOrEditFormProps {
  submitButtonName?: string;
  form: FormInstance;
  onFinish: (values: Record<string, any>) => Promise<void>;
}

const ConferenceCreateOrEditForm: React.FunctionComponent<ConferenceCreateOrEditFormProps> = ({
  submitButtonName = 'Create',
  form,
  onFinish,
}: ConferenceCreateOrEditFormProps) => (
  <Form
    initialValues={{
      type: ConferenceType.Venue,
      dateType: ConferenceDateType.Single,
    }}
    form={form}
    onFinish={onFinish}
  >
    <ConferenceBasicInfoForm />
    <ConferenceLocationForm />
    <ConferenceDateForm />
    <SubmitButton htmlType="submit" type="primary">
      {submitButtonName}
    </SubmitButton>
  </Form>
);

```

```

    </SubmitButton>
  </Form>
);

const SubmitButton = styled(Button)`
  margin-top: 40px;
  margin-left: 35px;
  height: 45px;
  width: 200px;
`;

export default ConferenceCreateOrEditForm;

import { Form, Input } from 'antd';
import { AlignRightOutlined } from '@ant-design/icons';
import React from 'react';
import { useTranslation } from 'react-i18next';

import FormLayout from '../components/FormLayout';

const ConferenceBasicInfoForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      title={t('Basic Info')}
      description={t('Event Description')}
      icon={<AlignRightOutlined />}
    >
      <Form.Item
        name="title"
        rules={[
          { required: true, message: t('Please provide conference title') },
        ]}
      >
        <Input placeholder={t('Enter conference name')} />
      </Form.Item>
      <Form.Item
        name="organizer"
        rules={[{ required: true, message: 'Please enter organizer info' }]}
      >
        <Input placeholder={t('Organizer Info')} />
      </Form.Item>
    </FormLayout>
  );
};

export default ConferenceBasicInfoForm;

import { Form, Input } from 'antd';

```

```

import React from 'react';
import { useTranslation } from 'react-i18next';
import { EnvironmentOutlined } from '@ant-design/icons';

import FormLayout from '../components/FormLayout';

const ConferenceLocationForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      title={t('Location')}
      description={t('Location Description')}
      icon={<EnvironmentOutlined />}
    >
      <Form.Item
        name="address"
        rules={[{ required: true, message: 'Please enter address' }]}
      >
        <Input placeholder={t('Enter address')} />
      </Form.Item>
    </FormLayout>
  );
};

export default ConferenceLocationForm;

import { Form, DatePicker, TimePicker } from 'antd';
import React from 'react';
import { useTranslation } from 'react-i18next';
import { FieldTimeOutlined } from '@ant-design/icons';
import styled from 'styled-components';

import FormLayout from '../components/FormLayout';

const ConferenceDateForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      title={t('Date and time')}
      description={t('Date Description')}
      icon={<FieldTimeOutlined />}
    >
      <Title>{t('Start and end times')}</Title>
      <Form.Item
        name="date"
        rules={[{ required: true, message: t('Date Required') }]}
      >
        <DatePicker.RangePicker

```

```

        placeholder={[t('Start Date'), t('End Date')]}
        autoComplete="off"
      />
    </Form.Item>
    <Form.Item
      name="time"
      rules={[{ required: true, message: t('Time Required') }]}
    >
      <TimePicker.RangePicker
        placeholder={[t('Start Time'), t('End Time')]}
        autoComplete="off"
        picker="time"
        format="HH:mm"
      />
    </Form.Item>
  </FormLayout>
);
};

const Title = styled.div`
  margin-top: 24px;
  margin-bottom: 12px;
  font-size: 0.875rem;
  font-weight: 600;
`;

export default ConferenceDateForm;

import { Form } from 'antd';
import React, { useEffect } from 'react';
import * as R from 'ramda';
import { useDispatch, useSelector } from 'react-redux';
import { useParams } from 'react-router-dom';

import OnboardingConferencePageView from '../components/OnboardingConferencePageView';
import { getConferenceById } from '../ConferenceReducer';
import { ConferenceOnboardingStep } from '../models/conference';
import {
  normalizeBasicToInitial,
  normalizeBasicConferenceToCreate,
  appendNewStep,
} from '../utils/conference-helpers';
import { updateConference } from '../ConferenceActions';

const OnboardingConferencePage: React.FunctionComponent = () => {
  const [form] = Form.useForm();
  const { conferenceId, onboardingStep } = useParams();

  const conference = useSelector((state: Record<string, any>) =>
    getConferenceById(state, conferenceId),

```

```

);

const dispatch = useDispatch();

const getInitialData = (
  onboardingStep: string,
  conferenceData: Record<string, any>,
): Record<string, any> => {
  switch (onboardingStep) {
    case ConferenceOnboardingStep.Basic: {
      return normalizeBasicToInitial(conferenceData);
    }
    case ConferenceOnboardingStep.Details: {
      return R.pick(
        ['image', 'summary', 'files', 'description'],
        conferenceData,
      );
    }
    default: {
      return null;
    }
  }
};

const getNormalizedData = (
  onboardingStep: string,
  values: Record<string, any>,
): Record<string, any> => {
  switch (onboardingStep) {
    case ConferenceOnboardingStep.Basic: {
      return normalizeBasicConferenceToCreate(values);
    }
    default: {
      return values;
    }
  }
};

useEffect(() => {
  if (form && Boolean(conference)) {
    const initialData = getInitialData(onboardingStep, conference);

    if (Boolean(initialData)) {
      form.setFieldsValue(initialData);
    }
  }
}, [form, conference, onboardingStep]);

const onFinish = async (values: Record<string, any>): Promise<void> => {
  await dispatch(

```

```

    updateConference({
      conferenceId,
      onboardedSteps: appendNewStep(
        onboardingStep,
        conference.onboardedSteps,
      ),
      ...getNormalizedData(onboardingStep, values),
    }),
  );
};

return R.isEmpty(conference) ? null : (
  <OnboardingConferencePageView
    onboardingStep={onboardingStep}
    form={form}
    onFinish={onFinish}
  />
);
};

export default OnboardingConferencePage;

import React from 'react';
import { FormInstance } from 'antd/lib/form';
import styled from 'styled-components';

import Layout from '@components/Layout';

import ConferenceCreateOrEditForm from '../forms/ConferenceCreateOrEditForm';
import ConferenceDetailForm from '../forms/ConferenceDetailForm';
import { ConferenceOnboardingStep } from '../models/conference';
import ConferenceInvitesContainer from '../containers/ConferenceInvitesContainer';

interface OnboardingConferencePageProps {
  onboardingStep: string;
  form: FormInstance;
  onFinish: (values: Record<string, any>) => Promise<void>;
}

const OnboardingConferencePage: React.FunctionComponent<OnboardingConferencePageProps> = ({
  onboardingStep,
  form,
  onFinish,
}: OnboardingConferencePageProps) => (
  <Layout>
    <FormWrapper>
      {onboardingStep === ConferenceOnboardingStep.Basic && (
        <ConferenceCreateOrEditForm
          form={form}
          onFinish={onFinish}

```

```

        submitButtonName="Update"
      />
    )}
    {onboardingStep === ConferenceOnboardingStep.Details && (
      <ConferenceDetailForm form={form} onFinish={onFinish} />
    )}
    {onboardingStep === ConferenceOnboardingStep.Invites && (
      <ConferenceInvitesContainer />
    )}
  </FormWrapper>
</Layout>
);

const FormWrapper = styled.div`
  margin: 0 15%;
  padding-bottom: 40px;

  input {
    height: 45px;
  }

  @media (max-width: 768px) {
    margin: 0 30%;
  }

  @media (max-width: 1024px) {
    margin: 0 10%;
  }
`;

export default OnboardingConferencePage;

import { Button, Form } from 'antd';
import { FormInstance } from 'antd/lib/form';
import React from 'react';
import { useTranslation } from 'react-i18next';
import styled from 'styled-components';

import ConferenceMainImageForm from './ConferenceMainImageForm';
import ConferenceDescriptionForm from './ConferenceDescriptionForm';
import ConferenceFilesForm from './ConferenceFilesForm';

interface ConferenceDetailFormProps {
  form: FormInstance;
  onFinish: (values: Record<string, any>) => Promise<void>;
}

const ConferenceDetailForm: React.FunctionComponent<ConferenceDetailFormProps> = ({
  form,
  onFinish,

```



```

}: ConferenceDetailFormProps) => {
  const { t } = useTranslation();

  return (
    <Form form={form} onFinish={onFinish}>
      <ConferenceMainImageForm />
      <ConferenceFilesForm />
      <ConferenceDescriptionForm />
      <SubmitButton htmlType="submit" type="primary">
        {t('Update')}
      </SubmitButton>
    </Form>
  );
};

const SubmitButton = styled(Button)`
  margin-top: 40px;
  margin-left: 35px;
  height: 45px;
  width: 200px;
`;

export default ConferenceDetailForm;

import { Form } from 'antd';
import { FileOutlined } from '@ant-design/icons';
import React from 'react';
import { useTranslation } from 'react-i18next';

import FormLayout from '../components/FormLayout';
import FilesUploadContainer from '../containers/FilesUploadContainer';

const ConferenceFilesForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      title={t('Files Title')}
      description={t('Files Desc')}
      icon={<FileOutlined />}
    >
      <Form.Item name="files">
        <FilesUploadContainer />
      </Form.Item>
    </FormLayout>
  );
};

export default ConferenceFilesForm;
import { Form } from 'antd';

```

```

import { PictureOutlined } from '@ant-design/icons';
import React from 'react';
import { useTranslation } from 'react-i18next';

import FormLayout from '../components/FormLayout';
import ImageUploadDraggerContainer from '../containers/ImageUploadDraggerContainer';

const ConferenceMainImageForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      icon={<PictureOutlined />}
      title={t('Main Conference Image')}
      description={t('Image Desc')}
    >
      <Form.Item name="image" valuePropName="value">
        <ImageUploadDraggerContainer />
      </Form.Item>
    </FormLayout>
  );
};

export default ConferenceMainImageForm;

import { Form, Input } from 'antd';
import { FontSizeOutlined } from '@ant-design/icons';
import React from 'react';
import { useTranslation } from 'react-i18next';

import RichTextEditor from '@components/RichTextEditor';

import FormLayout from '../components/FormLayout';

const ConferenceDescriptionForm: React.FunctionComponent = () => {
  const { t } = useTranslation();

  return (
    <FormLayout
      title={t('Description')}
      description={t('Description Detail')}
      icon={<FontSizeOutlined />}
    >
      <Form.Item
        name="summary"
        rules={[{ required: true, message: t('required') }]}
      >
        <Input placeholder={t('Description Placeholder')} />
      </Form.Item>
      <Form.Item name="description">

```

```

    <RichTextEditor isValid={true} minHeight={150} />
  </Form.Item>
</FormLayout>
);
};

export default ConferenceDescriptionForm;

import React, { useEffect, useState } from 'react';
import { useDispatch, useSelector } from 'react-redux';
import { useParams } from 'react-router-dom';

import PreviewPageView from '../components/PreviewPageView';
import { getPreviewConference } from '../ConferenceActions';
import { getConferenceById } from '../ConferenceReducer';

const PreviewPage: React.FunctionComponent = () => {
  const { conferenceId } = useParams();

  const [loading, setLoading] = useState(false);
  const conferenceData = useSelector((state: any) =>
    getConferenceById(state, conferenceId),
  );

  const dispatch = useDispatch();

  useEffect(() => {
    (async (): Promise<void> => {
      setLoading(true);

      await dispatch(getPreviewConference(conferenceId));

      setLoading(false);
    })();
  }, []);

  return <PreviewPageView conferenceData={conferenceData} loading={loading} />;
};

export default PreviewPage;

import { Spin, Row, Col } from 'antd';
import * as R from 'ramda';
import React from 'react';
import styled from 'styled-components';

interface PreviewPageViewProps {
  conferenceData: Record<string, any>;
  loading: boolean;
}

```

```

const PreviewPageView: React.FunctionComponent<PreviewPageViewProps> = ({
  conferenceData,
  loading,
}: PreviewPageViewProps) => (
  <div>
    {loading ? (
      <Spin spinning={loading} />
    ) : (
      <Wrapper>
        <ConferenceCardWrapper>
          <RowWrapper>
            <Col span={11} offset={3}>
              <Row>
                <ShortDate>
                  JUL
                  <br /> 14
                </ShortDate>
              </Row>
              <Row>
                <Title>{R.prop('title', conferenceData)}</Title>
              </Row>
              <Row>{R.prop('summary', conferenceData)}</Row>
              <Row>
                <div
                  dangerouslySetInnerHTML={R.prop(
                    'description',
                    conferenceData,
                  )}
                />
              </Row>
            </Col>
            <Col span={8}>
              <Row>
                Date And Time Tue, July 14, 2020 7:00 PM – 10:00 PM CEST
              </Row>
              <Row>Location Киссимми 155 5th street 30 Краков Poland</Row>
              <Row>
                Refund Policy Contact the organizer to request a refund. Sarpo
                fee is nonrefundable.
              </Row>
            </Col>
          </RowWrapper>
        </ConferenceCardWrapper>
      </Wrapper>
    )}
  </div>
);

const Wrapper = styled.div`

```

```

    min-height: 100vh;
    display: flex;
    justify-content: center;
    background-color: #f8f7fa;
  `;

const RowWrapper = styled(Row)`
  padding: 45px 0;
`;

const ShortDate = styled.div`
  color: #39364f;
  font-size: 0.93rem;
`;

const Title = styled.div``;

const ConferenceCardWrapper = styled.div`
  margin-top: 5%;
  width: 70%;
  box-shadow: 0 1px 2px 0 rgba(0, 0, 0, 0.15);
  background-color: #fff;
`;

export default PreviewPageView;

import { Form } from 'antd';
import React, { useState } from 'react';
import { useDispatch, useSelector } from 'react-redux';
import { useParams } from 'react-router-dom';

import ConferenceInvites from '../components/ConferenceInvites';
import ConferenceFormModal from '../modals/ConferenceFormModal';
import { sendInviteLink } from '../ConferenceActions';
import { getConferenceInvites } from '../ConferenceReducer';

const ConferenceInvitesContainer: React.FunctionComponent = () => {
  const { conferenceId } = useParams();

  const [form] = Form.useForm();
  const [isVisibleCreateMode, setIsVisibleCreateModel] = useState(false);
  const [loading, setLoading] = useState(false);
  const currentInvites = useSelector((state: any) =>
    getConferenceInvites(state, conferenceId),
  );

  const dispatch = useDispatch();

  const onFinish = async (values): Promise<void> => {
    setLoading(true);

```

```

    await dispatch(sendInviteLink({ conferenceId, ...values }));

    setLoading(false);
  };

  return (
    <
      <ConferenceFormModal
        loading={loading}
        onFinish={onFinish}
        form={form}
        isVisible={isVisibleCreateMode}
        onCancel={() => setIsVisibleCreateModel(false)}
      />
      <ConferenceInvites
        invites={currentInvites}
        onOpenInvitingModal={() => setIsVisibleCreateModel(true)}
      />
    </>
  );
};

export default ConferenceInvitesContainer;

import { Form } from 'antd';
import React, { useState } from 'react';
import { useDispatch, useSelector } from 'react-redux';
import { useParams } from 'react-router-dom';

import ConferenceInvites from '../components/ConferenceInvites';
import ConferenceFormModal from '../modals/ConferenceFormModal';
import { sendInviteLink } from '../ConferenceActions';
import { getConferenceInvites } from '../ConferenceReducer';

const ConferenceInvitesContainer: React.FunctionComponent = () => {
  const { conferenceId } = useParams();

  const [form] = Form.useForm();
  const [isVisibleCreateMode, setIsVisibleCreateModel] = useState(false);
  const [loading, setLoading] = useState(false);
  const currentInvites = useSelector((state: any) =>
    getConferenceInvites(state, conferenceId),
  );

  const dispatch = useDispatch();

  const onFinish = async (values): Promise<void> => {
    setLoading(true);

```

```

    await dispatch(sendInviteLink({ conferenceId, ...values }));

    setLoading(false);
  };

  return (
    <
      <ConferenceFormModal
        loading={loading}
        onFinish={onFinish}
        form={form}
        isVisible={isVisibleCreateMode}
        onCancel={() => setIsVisibleCreateModel(false)}
      />
      <ConferenceInvites
        invites={currentInvites}
        onOpenInvitingModal={() => setIsVisibleCreateModel(true)}
      />
    </>
  );
};

export default ConferenceInvitesContainer;

import { Form, Input } from 'antd';
import { FormInstance } from 'antd/lib/form';
import React from 'react';
import { useTranslation } from 'react-i18next';

import { SubmitButton } from './SignUpForm';
const SignInForm: React.FunctionComponent<SignInFormProps> = ({
  form,
  onFinish,
}: SignInFormProps) => {
  const { t } = useTranslation();

  return (
    <Form form={form} name="signIn" onFinish={onFinish}>
      <Form.Item
        name="email"
        rules={[
          {
            required: true,
            message: t('required'),
          },
        ]
      />
      <Input placeholder={t('Enter your email')} />
    </Form.Item>
    <Form.Item

```

```

    name="password"
    rules={[
      {
        required: true,
        message: t('required'),
      },
    ]}
  >
    <Input type="password" placeholder={t('Enter your password')} />
  </Form.Item>
  <SubmitButton htmlType="submit" type="primary">
    {t('signin')}
  </SubmitButton>
</Form>
);
};
export default SignInForm;
import { Form, Input, Button } from 'antd';
import { Store } from 'antd/lib/form/interface';
import { FormInstance } from 'antd/lib/form';
import React from 'react';
import { useTranslation } from 'react-i18next';
import styled from 'styled-components';
import PasswordBar from '@components/PasswordStrengthBar';
interface SignUpFormProps {
  form: FormInstance;
  passwordToMeasure: string;
  setPasswordToMeasure: Function;
  validatePassword: (_: undefined, password: string) => void;
  onFinish: (values: Store) => void;
}
const SignUpForm: React.FunctionComponent<SignUpFormProps> = ({
  form,
  passwordToMeasure,
  setPasswordToMeasure,
  validatePassword,
  onFinish,
}: SignUpFormProps) => {
  const { t } = useTranslation();
  return (
    <Form form={form} name="signUp" onFinish={onFinish}>
      <Form.Item
        name="email"
        validateTrigger="onBlur"
      >
        <Input type="email" placeholder={$ {t('Enter your email')}}... />
      </Form.Item>
      <Form.Item
        name="name"
        rules={[{ required: true, message: t('required') }]}

```



```

    validateTrigger="onBlur"
  >
    <Input placeholder={${t('Enter your name')}}... />
  </Form.Item>
  <Form.Item
    name="password"
    normalize={(value: string | undefined): string => {
      setPasswordToMeasure(value);
      return value;
    }}
  >
    <Input type="password" placeholder={${t('Enter your password')}}... />
  </Form.Item>
  <PasswordBar password={passwordToMeasure} />
  <SubmitButton htmlType="submit" type="primary">
    Sign Up
  </SubmitButton>
</Form>
);
};
export const SubmitButton = styled(Button)`
  margin-top: 20px;
  width: 100%;
  height: 45px;
  font-size: 1rem;
`;

export default SignUpForm;

```

ДОДАТОК В

Програмна платформа для організації міжнародних конференцій

Опис програми

УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_20Б

Аркушів 2

Київ 2020

АНОТАЦІЯ

Даний додаток містить опис двох основних модулів розроблених для програмної платформи. Створені модулі показують взаємодію сервера та клієнта та виконують такі завдання:

- введення даних;
- наормалізація даних в необхідний для системи вигляд;
- обробка даних;
- відображення даних.

При розробці обох модулів платформи використовувалась мова Typescript та NodeJs у середовищі написання коду — Microsoft Visual Studio Code.

ЗМІСТ

| | |
|---|----|
| 1. Загальні відомості..... | 78 |
| 2. Функціональне призначення | 79 |
| 3. Опис логічної структури | 80 |
| 4. Технічні засоби, що використовуються | 81 |
| 5. Вхідні і вихідні дані..... | 82 |

ЗАГАЛЬНІ ВІДОМОСТІ

У цьому додатку міститься опис програмної платформи для організації міжнародних конференцій. У додатку Б міститься програмний код основних модулів розроблюваної програмної платформи.

Розроблений додаток може працювати кросс-платформенно але потребує додаткового встановлення на пристрій сервіс Docker.

При розробці програмної платформи використовувалась мова TypeScript та NodeJs з використанням Microsoft Visual Studio Code середовища.

ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ

Розроблений додаток виконує завдання планування та організації міжнародних конференцій. За допомогою розробленої платформи можна легко і просто створити нову конференцію. Відправити запрошення будь якому користувачу на пошту. Відображення детальної інформації про конференцію. Функціональні обмеження на використання компонентів полягають лише в кількості введення даних та завантажених файлів.

ОПИС ЛОГІЧНОЇ СТРУКТУРИ

Програмна платформа складається з клієнтської частини та серверної частини. Взаємодія клієнта та сервера відбувається звичайним HTTP протоколом.

Загальний принцип роботи додатку наступний:

- 1) Користувач відправляє дані для створення конференції на сервер методом POST.
- 2) Нова конференція створюється та повертається на клієнт.
- 3) Користувач натискає на елемент списку конференцій для детального перегляду.
- 4) На сервер відправляється запит методом GET.
- 5) MongoDB запитом знаходиться відповідний документ.
- 6) Результат запиту повертається на клієнт.
- 7) Користувач натискає на кнопку для запрошення учасника конференції.
- 8) На сервер відправляється запит методом POST.
- 9) Адреса отримувача передається на сервіс для відправлення поштового повідомлення та результат повертається на клієнт.

ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ

Для користування програмною платформою потрібен веб-браузер останніх версій (на будь-якій доступній операційній системі) Google Chrome, Mozilla Firefox, Opera, Safari. Розробка і тестування під інші браузери не проводилась.

Також реалізовано адаптивний підхід до відображення змісту.

ВХІДНІ І ВИХІДНІ ДАНІ

Вхідні дані для веб-порталу: — текстова інформація, файли; Вихідні дані для веб-порталу: — записи в таблиці бази даних;

ДОДАТОК Г

Програмна платформа для організації міжнародних конференцій

АПРОБАЦІЯ

Міжнародна науково-практична конференція «Сучасні проблеми наукового забезпечення енергетики: Матеріали XVIII Міжнародної науково-практичної конференції молодих вчених і студентів 2020 року», м. Київ, 2020

УКР.НТУУ «КПІ ім. Ігоря Сікорського»_ТЕФ_АПЕПС_ТМ62195_20Б

Аркушів 4

2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали XVIII Міжнародної
науково-практичної конференції
молодих вчених і студентів
2020 року

ТОМ 2



Київ- 2020

| | |
|---|-----|
| Smart pointers for filling and sorting arrays. <i>STRELNIKOV N.P., student gr. TP-81</i> <i>Scientific chief - assoc.prof., cand.phys.-math.sc. Karpenko S.G.</i> | 110 |
| Moving constructor and relocation operator. <i>MORDAS I.S., student gr. TP-81</i> <i>Scientific chief - assoc.prof., cand.phys.-math.sc. Karpenko S.G.</i> | 111 |
| СЕКЦІЯ №10 МОДЕЛЮВАННЯ ТА АНАЛІЗ ТЕПЛОЕНЕРГЕТИЧНИХ ПРОЦЕСІВ | |
| | 112 |
| Розвантаження каналу передачі даних при real-time синхронізації даних датчиків та серверу в системах з використанням C++. <i>СКОРОБОГАТСЬКИЙ Д.В., магістрант гр. ТІ-91мн</i> <i>Керівник - доц., к.т.н. Кузьменко І.М.</i> | 113 |
| Інтелектуальний агент моніторингу та управління енергетичними системами. <i>ПИРОГОВСЬКА Т.В., магістрант гр. ТІ-91мн</i> <i>Керівник - доц., к.т.н. Ковальчук А.М.</i> | 114 |
| Програмні засоби підсилення інтенсивності гідроакустичних сигналів. <i>ОБРУСНИК Д.В., магістрант гр. ТВ-91мн</i> <i>Керівник - доц., к.т.н. Кублій Л.І.</i> | 115 |
| Модель даних для створення інженерних мереж студмістечка. <i>КОЗАЧУК О.В., студент гр. ТМ-61</i> <i>Керівник - асист. Швайко В.Г.</i> | 116 |
| Web-сервіс з передбаченням жанру фільмів. <i>ЄРОХІНА А.О., студент гр. ТВ-61</i> <i>Керівник - доц., к.т.н. Кублій Л.І.</i> | 117 |
| Веб-платформа для організації міжнародних конференцій. <i>ГЛАДКИЙ О.Л., студент гр. ТМ-62</i> <i>Керівник - доц., к.т.н. Кублій Л.І.</i> | 118 |
| Інтелектуальний агент системи контролю та управління доступом. <i>ВИСОВЕНЬ Д.Д., студент гр. ТІ-61</i> <i>Керівник - доц., к.т.н. Ковальчук А.М.</i> | 119 |
| Система надання статистики про інженерні вакансії в ІТ. <i>БОЧОК В.О., студент гр. ТВ-61</i> <i>Керівник - доц., к.т.н. Кублій Л.І.</i> | 120 |
| Класифікація знімків для аналізу часових змін лісових насаджень. <i>БОГАЧ А.Г., студент гр. ТМ-62; БАБ'ЯК В.В., студент гр. ТМ-62</i> <i>Керівник - асист. Швайко В.Г.</i> | 121 |
| Horizontal Autoscaling of Microservices in a Cloud-based Kubernetes Cluster. <i>VOINALOVYCH V.A., student gr. ТІ-62</i> <i>Scientific chief - assoc.prof., cand.phys.-math.sc. Smakovskiy D.S.</i> | 122 |
| Microservice conservation profile of the university unit. <i>PONOSHOVNA O.O., student gr. TP-62</i> <i>Scientific chief - assoc.prof., cand.eng.sc. Smakovskiy D.S.</i> | 123 |
| Robotic platforms in IOT Environments. <i>HOLETS V.O., student gr. ТІ-61</i> <i>Scientific chief - assoc.prof., cand.eng.sc. Kovalchuk A.M.</i> | 124 |

УДК 004.738.5

Студент 4 курсу, гр. ТМ-62 Гладкий О.Л.
Доц., к.т.н. Кублій Л.І.

ВЕБ-ПЛАТФОРМА ДЛЯ ОРГАНІЗАЦІЇ МІЖНАРОДНИХ КОНФЕРЕНЦІЙ

Програмне забезпечення для організації конференцій або інших заходів — це засіб, який допомагає організаторам досягнути успіху в усіх аспектах — від реєстрації і планування заходів до створення звітів. Оскільки програмне забезпечення для керування заходами об'єднує всі інструменти в одну платформу, то це робить програму легкою для розуміння і надає організаторові велику кількість готових ідей [1].

Планування конференцій є складним завданням. Потрібно визначити дату конференції, скласти інформаційний лист, визначити і оформити місце проведення конференції, зібрати заявки потенційних учасників, тексти їхніх доповідей, презентації, інший матеріал, інформацію про оплату участі, спілкуватися з учасниками, розмістити різноманітні оголошення, сформулювати за інформацією з заявок програму, розмістити її на сайті і розіслати учасникам, розмістити на сайті звіт конференції, включаючи фотографії й відео-матеріали, збірник матеріалів конференції. Наявність при цьому веб-платформи для організації конференцій робить процес швидким і простим для організаторів, а також доступним для учасників конференції і всіх зацікавлених осіб.

До появи програмного забезпечення для управління заходами організаторам потрібно було використовувати сторонні інструменти такі, як WordPress чи Squarespace, для власноручного проектування сайтів заходів. У наш час існує досить багато веб-додатків для планування заходів, наприклад, Meeting Play, Event Integrity, Event Core, Planning Pod, ClearEvent та інші. Вони мають різний функціонал, спрямований на оптимізацію процесу керування заходами, і виключають необхідність ручного створення власних сайтів [2]. У цілому, сучасне програмне забезпечення для керування й організації конференцій робить більше, ніж просто відображає захід на сайті. Воно дає можливість організаторам розкрити всі деталі подій і, в результаті, забезпечити успіх заходу.

У розробленій платформі впроваджено функціонал існуючих веб-додатків, складовими якого є авторизація, реєстрація, редагування даних про користувача, можливість збирати різноманітні матеріали, переглядати матеріали поточної і попередніх конференцій. Також є можливість організатору створювати будь-яку кількість запрошень, виконати їхнє редагування, видалення. Користувач при створенні або редагуванні заходу може прикріпити файли різного типу, зображення, опис, карту місцезнаходження, вказати час і дату проведення заходу. На головній сторінці веб-платформи виконується фільтрація за датою проведення заходу. Картка конференції містить таку інформацію, як назва, місце, дата і час, опис, файли будь-якого формату, зображення, надає можливість зареєструватися. На головній панелі є посилання на сторінку редагування персональних даних.

Створений веб-додаток має три компоненти: головна сторінка для перегляду конференцій, головна панель для їхнього створення і модуль для редагування персональних даних користувачів. При розробці модулів веб-платформи для написання клієнтської частини вибрано мову Typescript і веб-фреймворк ReactJS, для серверної — NodeJS, також використано мову запитів і маніпулювання даними GraphQL, базу даних MongoDB.

Розроблена веб-платформа допоможе спростити планування конференцій і надасть інструменти, необхідні для створення й подальшого супроводу таких заходів.

Перелік посилань:

1. Why you need event management software. — <https://blog.bizzabo.com/why-you-need-event-management-software>
2. Best event management software — zapier.com/blog/best-event-management-software